FAIRCHILD SEMICONDUCTOR

Do NOT Sell

MADE IN
FAIRCHILD

THE TTL APPLICATIONS HANDBOOK

AUGUST 1973

THE TTL APPLICATIONS HANDBOOK

Prepared by
the Digital Applications Staff
of Fairchild Semiconductor

Edited by
Peter Alfke and Ib Larsen

FAIRCHILD
SEMICONDUCTOR

464 Ellis Street, Mountain View, California 94042

Fairchild cannot assume responsibility for use of any circuitry described other than circuitry entirely embodied in a Fairchild product. No other circuit patent licenses are implied

Preface

In the last five years, the art of digital system and logic design has undergone dramatic changes. Integrated circuits of increasing complexity have not only become available but also cost competitive, and the classical rules of logic design have lost their significance. It is no longer essential to minimize the number of gates or flip-flops in order to optimize a design. The more meaningful task of minimizing total system cost is now the primary design consideration.

There is a gap between the classical rules of logic design found in textbooks and taught in universities, and the real-world requirements of cost effective system and logic design using currently available components. This book provides many ideas and suggestions covering efficient use of TTL/MSI. Most of the concepts are readily applicable to ECL or CMOS designs.

The INTRODUCTION, written by Peter Alfke, explains the impact of MSI circuits on system design and how gate minimizing techniques have given way to imaginative use of complex functional components. General system design rules are offered and the advantages as well as the constraints of MSI are outlined.

MULTIPLEXERS are covered in Chapter 1. Eric Breeze and Peter Alfke explain their use in data routing, bussing and control sequencing, cover such applications as pattern generation and keyboard encoding, and show how multiplexers can perform efficiently as function generators of highly irregular functions. In the next chapter, Eric Breeze goes on to explain DECODERS and such applications as memory address, data or clock demultiplexing, minterm generation and switch encoding.

DIGITAL DISPLAY SYSTEMS, including incandescent, fluorescent, gas discharge and LED displays and decoder/driver circuits are explained by Eric Breeze in the third chapter. Many detailed display multiplexing circuits are included. In the ENCODER chapter, Bob Montevaldo describes the operation of priority encoders and their use in such areas as hardware priority interrupt, switch encoding, D/A conversion and rate multiplication.

Arithmetic OPERATOR circuits and systems performing addition, subtraction, multiplication and division, comparison, parity check, error correction and code conversion are described in Chapter 5 by Peter Alfke. The use of Zero and One, High and Low is clarified, negative numbers and carry look-ahead are discussed, and throughout the chapter the trade offs between fast parallel and simple serial methods are emphasized.

Chapter 6 covers LATCHES and applications such as holding registers and contact bounce eliminators. The 8-bit addressable latch is covered in detail and many converter, demultiplexing and filter applications are included. The author, Bob Montevaldo, goes on to Chapter 8, REGISTERS, and discusses their operating characteristics and applications as register files, recirculating memories, a pseudo-random sequence generator, and a typical minicomputer Arithmetic Logic Unit.

MEMORY circuits are described in Chapter 7, including random access read/write and read only memory devices. A detailed discussion of timing requirements for read/write memory circuits, a Last In/First Out memory, and methods for changing and expanding ROMs are included.

COUNTERS by Peter Alfke provides an overview of asynchronous, semisynchronous, presettable and bidirectional MSI counters. Many applications such as the use of counters in divider configurations, high speed presettable synchronous counters, shift registers as counters, an industrial counter and a digital stopwatch are covered.

SMALL SCALE INTEGRATION is the subject of Chapter 10, in which Peter Aflke describes gates, inverters and buffers and such applications as oscillators and edge detectors. The second part of the chapter covers single and dual flip-flops and their applications as counters, differentiators, bounce eliminators, and switch synchronizers. Conventional applications of gates and flip-flops are not included because they are quite well-known and MSI has made them less important.

Chapters 11 and 12 cover LOW POWER and SCHOTTKY TTL respectively, and summarize their advantages and design precautions. Chapter 13, by Ken True, discusses MONOSTABLE MULTIVIBRATORS. Because they are half analog, half digital devices, circuit oriented descriptions are included explaining triggering and timing, retriggerable and non-retriggerable operation of both single and dual monostables, and the use of electrolytic capacitors. Applications include a frequency discriminator, pulse width detector, contact bounce eliminator, oscillator, and a gated clock generator.

In the next chapter, INTERFACE circuits are covered by Ken True. The important design considerations relevant to interface of TTL with transmission lines are explained, including the various aspects of single-ended vs. balanced differential methods, simplex vs. multiplex lines, and line matching. A technique for estimating signal quality as a function of line length is explained.

The final chapter is a summary of TTL ELECTRICAL CHARACTERISTICS. Peter Alfke describes input and output configurations, current and voltage levels, and thermal characteristics. General recommendations as to system layout, interconnections and decoupling are made. Timing aspects of edge-triggered devices are explained and problems associated with clock skew are discussed. Also included are various interfaces with other integrated circuit logic families and discrete devices.

This book is meant to be a complete "user's handbook" for TTL/MSI. It is based on ideas and designs generated in Fairchild's digital applications over the last five years. The authors acknowledge valuable ideas and contributions by Clive Ghest, John Nichols, Mogens Ravn and John Springer, the administrative backing of Bob Ulrickson, and the production coordination by Mary Jean Kulus and Marty Lindquist.

FAIRCHILD
SEMICONDUCTOR

Mountain View, California August 1973

The TTL Applications Handbook

Contents

Multiplexers 1
Decoders 2
Digital Display Systems 3
Encoders 4
Operators 5
Latches 6
Memories 7
Registers 8
Counters 9
TTL Small Scale Integration 10
Low Power TTL 11
Schottky TTL 12
Monostable Multivibrators 13
Transmission Line Interface 14
TTL Characteristics 15

TABLE OF CONTENTS

A Functional Cross Reference and a Product Index are located at the back of the book.

INTRODUCTION

The Impact of MSI on Logic Design IX
Advantages of MSI X
General System Design Rules X
TTL/MSI Device Design Considerations XI
User-Oriented TTL/MSI XII
Terminology XII

MULTIPLEXER

Selection Guide 1-2
Introduction 1-3
9322 Quad 2-Input Multiplexer 1-4
Data Routing 1-5
Dual 10-Input Multiplexer 1-6
Shift Left, Shift Right Parallel Load Register 1-6
Word Sorter for Comparing Four 4-Bit Words 1-7
9309 Dual 4-Input Multiplexer 1-8
93153/74153 Dual 4-Input Multiplexer 1-9
Dual 2-Input Multiplexer 1-10
Data Commutation 1-10
Multiple Word Data Bussing 1-10
9312/9318 8-Input Multiplexers 1-11
Time Multiplexing or Data Routing 1-12
Memory 1-13
8-Step Control Sequence 1-14
Decade Multiplexer 1-15
Digital 32-Input Multiplexer 1-15
93150/74150 16-Input Multiplexer 1-16
Pattern Generator 1-17
15-Key-to-Binary Code Converter 1-17
Multiplexers as Function Generators 1-18
Full Adder, Full Subtractor 1-19
X of Y Pattern Detector 1-19

DECODERS

Selection Guide 2-2
Introduction 2-3
9321 Dual 1-of-4 Decoder 2-4
1-of-64 Decoder 2-6
Demultiplexing 2-6
4-Phase Clock Generator 2-7
Function Generation 2-7
9301 and 9352/7442 Decoder 2-8
Address Decoding 2-9
Demultiplexing 2-10
Minterm Generation 2-11
Input-Output Table 2-11
9302 1-of-10 Decoder 2-12
A Scanning Thumbwheel Switch Encoder 2-13
9345, 9352, 9353, 9354 1-of-10 Decoders 2-14
Decoder Output States for Commonly Used Input Codes 2-14
9311 1-of-16 Decoder 2-15
Address Decoding 2-16
1-of-32 Decoder 2-16
Large Address Decoding 2-17
Read Only Memory Control 2-18
Read/Write Memory Control 2-19
Decoding and Encoding 2-20
Switch Encoding 2-20
Data Demultiplexing 2-21
Clock Demultiplexing 2-21
Single Master/Multiple Slave 2-22
Programmable Counter 2-22

DISPLAYS

Selection Guide 3-2
Introduction 3-3
Display Types 3-3
7-Segment Decoder/Drivers 3-4
7-Segment Decode Format 3-6
Zero Suppression 3-7
Multiplexing Systems 3-8
Demultiplexing System 3-9
A Simple, 8-Digit Multiplexed System 3-10
Multiplexed 8-Digit Display with Memory 3-11
8-Digit Display Using Shift Registers as a Recirculating Memory 3-12
Multiplexing 7-Segment Incandescent Displays 3-13
Fluorescent 7-Segment Displays 3-14
Driving Fluorescent Displays 3-15
Multiplexing Fluorescent Diode Displays 3-16
Multiplexing Fluorescent Triode Displays 3-17
Gas Discharge Digital Display Systems 3-18
Multiplexing Gas Discharge Displays 3-19
Light Emitting Diode Displays 3-20
Driving LED Displays 3-21
Multiplexing LED Displays 3-22
Multiplexed LED Calculator Display 3-23

ENCODERS

Selection Guide 4-2
Introduction 4-3
9318 8-Input Priority Encoder 4-4
Expanded Encoders 4-6
Decimal to BCD Encoders 4-7
Switch Encoder 4-8
Linear Priority Encoder 4-8
Code Conversion 4-9
Computer Priority Interrupts 4-10
Simultaneous Analog to Digital Conversion 4-11
D/A Conversions, Binary and BCD Rate Multipliers 4-13
Single Zero Detection 4-14

OPERATORS

Selection Guide 5-2
Introduction 5-3
Terminal Confusion 5-5
Carry Signals in Parallel Binary Adders 5-5
Carry Lookahead 5-6
Number Representation 5-6
Addition and Subtraction of Binary Numbers 5-7
Overflow 5-7

TABLE OF CONTENTS

Adders/Subtractors

Serial Binary Addition/Subtraction 5-9
Serial Addition of More Than Two Variables 5-10
5-Digit Bit-Serial Adder Subtractor (Accumulator) 5-11
9383/7483 4-Bit Ripple Carry Adder 5-12
The 9383 as a 3-Bit or a 2+1-Bit Adder 5-12
BCD Adder/Subtractor 5-13
16-Digit BCD Accumulator 5-14
Slow BCD Adder/Subtractor/Accumulator Using Counters 5-15
9340 Arithmetic Logic Unit 5-16
Signed Magnitude Addition and Subtraction 5-20
Complement Arithmetic 5-20
16-Bit Full Lookahead System 5-21
Comparison Functions 5-22
9341 Arithmetic Logic Unit 5-22
9342 Carry Lookahead 5-25
9341 Carry Methods 5-26
Arithmetic with the 9341 5-27
Comparison with the 9341 5-27
Arithmetic Register Systems 5-28
Single Address Arithmetic Register 5-28
3-Address Arithmetic Register 5-30
3-Address ALU with Eight Registers 5-31

Multipliers/Dividers

Multiplication/Division Using Counters 5-32
Fully Serial Binary Multiplier 5-33
8 by 8-Bit Binary Multiplier 5-34
16 by 16-Bit Binary Multiplier 5-35
Count and Shift BCD Multiplication 5-36
8-Bit Divider 5-38
9344 4 by 2-Bit Binary Multiplier 5-40
8 by 8-Bit Multiplication Array 5-41
16 by 16-Bit Multiplication Array 5-42
93S43 4 by 2-Bit Multiplier 5-43
8 by 4-Bit Binary Multiplier 5-44

Comparator Systems

Identity Comparators 5-45
Bit Serial Magnitude Comparison 5-46
Most Significant Bit First Magnitude Comparison 5-47
9324 5-Bit Comparator 5-48
Parallel Comparator Arrays 5-49

Error Detection/Correction

Parity Generation 5-51
Error Correction — Hamming Codes 5-51
9348 12-Input Parity Checker/Generator 5-53
Optimum Termination of Unused Inputs 5-54

Code Conversion

BCD-to-Binary Converter Using Adders 5-55
Serial In/Serial Out BCD-to-Binary Converter 5-56
Bit Serial Binary-to-BCD Converter 5-57
BCD-to-Binary Converter Using Counters 5-58
Binary Angle-to-BCD Converter 5-59
Gray Code Conversions 5-60
Excess 3 Gray Code 5-60
Generating Complements 5-61
Controlled Nines Complement Circuit 5-61

LATCHES

Selection Guide 6-2
Introduction 6-3
9308 Dual 4-Bit Latch 6-5
Register Selection 6-6
Register Matrix 6-7
Register Using Latches as M/S Flip-Flops 6-8
Single Master/Multiple Slave 6-8
Holding Register in Counting and Display 6-9
Multiplexing to a Data Bus 6-9
9314 4-Bit Latch 6-10
Operational Modes 6-11
Contact Bounce Eliminator 6-11
One or Zero Catching 6-11
Count and Hold Display System 6-12
Latch Selection 6-12
9334 8-Bit Addressable Latch 6-13
Addressable Latch Expansion 6-16
16-Word by 4-Bit Memory 6-17
64-Word by 4-Bit Memory 6-18
Demultiplexing and Decoding 6-19
Driving Discrete Devices 6-19
8-Bit A/D Converter 6-20
Programmable A/D Converter Logic 6-21
Serial-to-Parallel Conversion 6-22
4096-Bit, 15 MHz Serial Memory 6-23
Noise and Contact Bounce Suppression 6-24
9375, 9377 4-Bit Latches 6-25

MEMORIES

Selection Guide 7-2
Introduction 7-3
Memory Organization 7-3
Addressing Techniques 7-3
Memory Types 7-4
General Timing Considerations 7-4
Data Timing (Write Mode) 7-6
Address Timing (Write Mode) 7-6
Transition Time Uncertainties 7-6
Scratchpad Memory 7-7
Fast Timing Chart 7-8
Conventional Timing Chart 7-8
Interface 7-8
Scratchpad/File Applications 7-9
Micro-Control Storage Using Read/Write Memory 7-9
Buffer Memories 7-9
Main Memories 7-9
93410/93410A Random Access Memory 7-10
256-Word by 8-Bit Buffer Memory System 7-12
Last In/First Out (LIFO) Push Down Stack Memory 7-13
93415 Random Access Memory 7-14
Bipolar ROMs 7-15
MOS ROMs 7-15
Expanding ROMs 7-15
Changing the ROM Format 7-15
Selective Power Down 7-15
93434 Read Only Memory 7-16

TABLE OF CONTENTS

93406 Read Only Memory ... 7-17
Expanding ROMs — Increasing the Number of Outputs ... 7-18
Expanding ROMs — Increasing the Number of Addresses ... 7-18
Changing the ROM Format — More Addresses, Fewer Outputs ... 7-19
Changing the ROM Format — More Outputs, Fewer Addresses ... 7-20

REGISTERS

Selection Guide ... 8-2
Introduction ... 8-3
9300 Shift Register ... 8-6
Mode Selection/Data Entry ... 8-7
Quad D Flip-Flop or Dual 2-Bit Register ... 8-8
Left/Right Shift Register ... 8-8
Serial-to-Parallel Conversion ... 8-9
Dual Input Shift Register ... 8-10
Shift/Hold Capability ... 8-10
Recirculation ... 8-10
Edge Detector with Storage ... 8-10
Parallel-to-Serial Conversion ... 8-11
93164 8-Bit Serial to Parallel Converter ... 8-12
93165 8-Bit Parallel to Serial Converter ... 8-13
9328 Dual 8-Bit Shift Register ... 8-14
Recirculation ... 8-15
Data Enable ... 8-15
Minimum Input Pattern/Two Variables ... 8-15
Data Input Selection ... 8-16
Shift Register Selection ... 8-16
Serial Memories ... 8-17
Variable Length Shift Register ... 8-18
Multiplexed 7-Segment Count and Display Circuit ... 8-20
Modulo 16 Counter ... 8-21
Pseudo-Random Sequence Generators ... 8-21
9338 8-Bit Multiple Port Register ... 8-22
Equivalent Implementation of Four 9338s ... 8-24
Serial Expansion Capabilities ... 8-25
Parallel Expansion ... 8-26
Arithmetic Logic Unit ... 8-27
First In/First Out Memory ... 8-28
93H72 High Speed 4-Bit Shift Register with Enable ... 8-29
9391 8-Bit Shift Register ... 8-31
9394 4-Bit Shift Register ... 8-32
9395 4-Bit Shift Register ... 8-33
9396 5-Bit Shift Register ... 8-35

COUNTERS

Selection Guide ... 9-2
Introduction ... 9-3
9305 Variable Modulo Counter ... 9-6
Synchronous Binary Count ... 9-8
Standard 4-Bit Counter Configurations ... 9-8
50% Duty Cycle Output Dividers ... 9-9
Asynchronous Mode of the 9305 ... 9-9
Multistage Counters/Dividers ... 9-10
Count Modulo Control ... 9-10
Odd Modulo Dividers ... 9-11
Counter with Built-in Bounce Eliminator ... 9-14
Single/Multiple Count Enables ... 9-14
Counters Dividing by 45 and 50 ... 9-14
Basic TV Sync Generators ... 9-15
Synchronous Counter/Sequencer ... 9-16
A 2-Digit BCD Rate Multiplier ... 9-16
9310 and 9316 Counters ... 9-17
Mode Selection ... 9-19
Count Sequence ... 9-19
Multistage Counting ... 9-20
Programmable Divider ... 9-21
Rotary Switch Programming ... 9-21
Generating Complements ... 9-21
Counter Expansion ... 9-22
50% Duty Cycle Output Counters ... 9-22
Multistage Programmable Counters ... 9-23
Self-Stopping Counter ... 9-26
Counter with Three Enables ... 9-26
High Speed Programmable Divider ... 9-26
Resynchronizer ... 9-26
Programming Counters ... 9-27
Combination Counter/Shift Registers ... 9-27
Programmable Counters Starting at Zero ... 9-28
Cyclic D/A Conversion ... 9-29
9360, 9366 Up/Down Counters ... 9-30
Operating Modes and Count Sequences ... 9-30
Terminal Count Logic ... 9-32
Counter Cascading ... 9-32
Light-Controlled Up/Down Counting ... 9-33
Programmable Divider ... 9-33
Dead End Counters ... 9-34
Synchronization and Coincident Pulse Prevention ... 9-34
Single Line Up/Down Control ... 9-34
Counting with Shift Registers ... 9-35
Twisted Ring Reversible Counters ... 9-35
Linear Feedback Shift Counters ... 9-36
Feedback Shift Registers ... 9-36
Modulo 15 Counter ... 9-37
Up/Down Counters ... 9-37
Variable Modulo Counters ... 9-37
Multistage Divide-by-n Counters ... 9-38
Up/Down Counter with Display Multiplexer ... 9-39
Crystal Controlled Stopwatch with Multiplexed Display ... 9-40

SMALL SCALE INTEGRATION

Gate, Inverter and Buffer Selection Guide ... 10-2
Introduction ... 10-3
Collector-ANDed DTL Gates ... 10-3
Increasing the Voltage Swing of TTL Outputs ... 10-3
Edge Detector ... 10-4
Dual Edge Detector ... 10-4
Simple RC Clock Generator ... 10-5
Inexpensive Wide Range VCO ... 10-5
Single and Dual Flip-Flops ... 10-6
Clocking Scheme ... 10-6
Single and Dual Flip-Flop Selection Guide ... 10-7
Johnson (Moebius) Counters ... 10-8
Switch Bounce Eliminator and Digital Differentiator ... 10-8
Switch Synchronizer ... 10-8

TABLE OF CONTENTS

LOW POWER TTL

Selection Guide 11-2
Introduction 11-3

SCHOTTKY SUPER HIGH SPEED TTL

Selection Guide 12-2
Introduction 12-3
Advantages of Schottky TTL 12-4
Disadvantages of Schottky TTL 12-4
General Recommendations 12-4

MONOSTABLE MULTIVIBRATORS

Selection Guide 13-2
Introduction 13-3
Triggering Conditions 13-3
Non-Retriggerable Operation 13-4
Timing Equations and Nomographs 13-4
9600, 9601, 9602 Timing Nomograph 13-5
9603 Timing Nomograph 13-5
Typical Characteristics 13-6
Use of Electrolytic Timing Capacitors 13-7
Timing Networks 13-7
9601 Retriggerable Monostable Multivibrator 13-9
Timing Waveforms 13-12
Programmable Delay Time — PDM Generator 13-13
Pulse Duration Detector 13-13
Frequency Discriminator 13-13
Double Pulse Detection 13-13
Schmitt Trigger 13-14
Malfunction Indicator 13-14
Unstoppable Oscillator 13-14
Gated Clock Generator 13-14
Voltage Controlled Monostable 13-15
Form "A" (SPST) Contact Bounce Eliminator 13-15
9600 Single Monostable Multivibrator 13-16
9602/96L02 Dual Monostable Multivibrator 13-17
Dual Edge Triggering 13-20
Delayed Pulse Generation 13-20
Keyed Oscillator 13-20
Keyed or Synchronized Oscillator with Constant Output Pulse Width 13-20
9603/74121 Monostable Multivibrator 13-21
74122 Retriggerable Monostable Multivibrator 13-24
74123 Dual Retriggerable Monostable Multivibrator 13-25

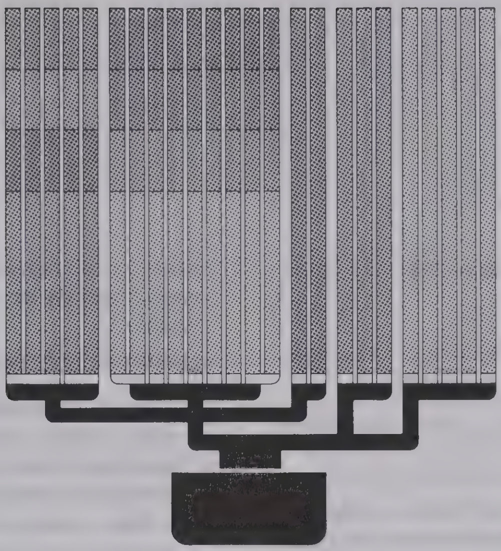
TRANSMISSION LINE INTERFACE ELEMENTS

Selection Guide 14-2
Introduction 14-3
Forms and Modes of Operation 14-3
Line Matching 14-5
Parallel Termination 14-6
Series Termination 14-6
Signal Quality on Long Transmission Lines 14-7
Binary Eye Pattern 14-8
Data Signal Quality 14-10
Selecting Line Drivers and Line Receivers 14-11
Interface Circuits 14-12
IBM 360/370 I/O Interface 14-13
EIA RS 232-C Interface with Failsafe Receivers 14-14
MIL STD-188C Interface 14-14

TTL CHARACTERISTICS

Introduction 15-3
Electrical Properties 15-4
Outputs 15-5
TTL Input and Output Impedance 15-6
Supply Voltage and Temperature Range 15-6
Input and Output Voltages 15-6
Input and Output Currents 15-7
DC Noise Margins 15-7
AC Noise Margins 15-8
Transmission Line Drive Capability 15-9
Ringing 15-9
Physical System Considerations 15-10
Transmission Line Effects 15-10
Crosstalk 15-11
Power Supply for TTL Systems 15-11
Decoupling 15-11
Grounds 15-11
Unused Inputs 15-11
Unused Gates 15-12
Extenders 15-12
Increasing Fan Out 15-12
Line Driving and Receiving 15-12
Flip-Flops and Registers 15-13
Propagation Delay 15-13
Set-up Time 15-13
One Catching Master/Slave Flip-Flops 15-14
Clock Skew 15-14
Interfacing 15-15
Interfacing TTL and DTL 15-15
Interfacing TTL and ECL 15-15
Interfacing TTL and CTL 15-16
TTL Driving Transistors 15-17
Shifting a TTL Output to Negative Levels 15-17
Transistors Driving TTL 15-17

A designer was feeling frustrations,
Gates and Flip-Flops were taxing his patience.
So he said: What a mess!
I can do it for less,
If I use Fairchild's smart applications.



INTRODUCTION

The ultimate objective of any digital system design is to achieve a given performance for the lowest total system cost. Total system cost includes design, tooling, manufacturing, and service. The relative importance of these factors varies with different industries. In aerospace, design and service tend to dominate. In consumer fields, manufacturing cost dominates. In any case, however, total cost over the life of the system should be minimized.

A system and logic designer achieves this goal with the selection of the appropriate system architecture and the selection of those components which are best for the system architecture. Today, there is an almost overwhelming variety of alternatives. In only ten years, several integrated circuit technologies have vied for the attention of designers.

- RTL, the earliest family, offered high performance but insufficient noise margins.
- DTL, slower but easier to use, offered improved noise margins and fan out. It has been gradually superceded by its faster cousin, TTL.
- TTL offers higher speed, better noise immunity and drive capability, by far the largest number of standard complex devices and various speed/power trade offs.
- CTL offers higher speed at moderate cost and power dissipation but requires more experience on the part of the user because the logic levels are non-restoring.
- ECL offers the highest speed, drives terminated transmission lines, and is the ultimate choice for very fast systems.
- MOS offers greatly increased complexity and lower power consumption but significantly less speed. It is predominately used in large regular arrays such as serial and random access memories, custom LSI circuits for high volume production — notably calculators, and microprocessors which implement logic with programming steps rather than hard wired connections.
- CMOS offers extremely low power consumption when operated at low speed. Complexity is comparable to TTL, speed is between MOS and TTL.

Although the applications in this book concentrate on TTL components, many are applicable to other technologies, particularly ECL and CMOS.

THE IMPACT OF MSI ON LOGIC DESIGN

In the days of vacuum tubes, transistors, diodes, and even integrated circuits containing gates and flip-flops (pre 1967), the art of logic design was clearly defined and measured. The logic designer attempted to design with a minimum number of components, using such established techniques as Karnaugh maps, Veitch diagrams, and Boolean algebra. System design, logic design and component selection were independent, requiring little interaction on the part of the designers. Medium Scale Integration, standard circuits with 20 to 100-gate complexity, has radically changed this relationship and made system design, logic design, and component selection heavily interdependent, each influencing and influenced by the others. It is no longer sufficient — or even important —to minimize the number of gates and flip-flops. It is far more important to select the proper complex integrated circuit which can perform the desired function most economically. It may even be appropriate to *redefine* subsystems to accommodate more sophisticated and more cost-effective components. In addition, the higher levels of integration also offer reduced power consumption and improved system reliability. Logic design is no longer an isolated art, has left its ivory tower and is more demanding, but at the same time far more stimulating and rewarding.

The logic designer today must be involved in system design, must know about the complex components available, and must be aware of economical impacts of semiconductors, printed circuit boards, connections, and power supplies. This knowledge, and the trade-offs represented, are necessary to achieve this goal . . . the lowest system cost for specified performance.

INTRODUCTION

ADVANTAGES OF MSI

Complex MSI devices offer many advantages over SSI gates and flip-flops.

- Increased packing density — more functions on a printed circuit board — simplified mechanical construction
- Reduced interconnections — solder joints, backplane wiring, and connectors
- Improved system reliability. Mean time between failure of an MSI is roughly the same as MTBF of an SSI device, so a reduction in package count increases system MTBF. Moreover, the reduced interconnections improved reliability.
- Reduced power consumption and total heat generation. However, the increased density possible with MSI can result in higher power and heat *densities.*
- Lower cost. MSI/SSI cost comparisons must consider true total cost, not only the purchase price of the integrated circuits. An unavoidable overhead cost is associated with each IC, attributed to testing, handling, insertion, and soldering plus the appropriate share of connectors, PC boards, power supplies, cabinets, etc. This overhead cost is generally estimated to be $.50 to $1.50 per circuit. When this cost is added to semiconductor cost, MSI offers more economical solutions even in cases when the MSI components are more expensive.
- Decreased design, debugging, and servicing costs and time. Because MSI devices are functional subsystems, it is much easier and faster to design a system with them. This is an important consideration when average product design lifetime shrinks and designers are pressed to complete increasingly complex designs in less and less time. Functional partitioning also simplifies debugging and service.

Because of these obvious advantages, MSI is generally accepted in the regular and repetitive portions of digital designs. Today, no one would build a synchronous counter with dual flip-flops or a latch array with NAND gates. In less regular and less repetitive areas, notably control circuits, MSI has not yet found such acceptance. Instead, the old, gate-minimized approach using SSI devices prevails. To a large extent this is only because designers are not sufficiently aware of the more subtle features of good MSI circuits and the logic manipulations possible. MSI devices can be used to advantage even in very irregular and very specialized design areas. Moreover, the control features of synchronous counters and registers can significantly reduce specialized control logic.

Imaginative application of MSI components reduces the role of gates and flip-flops to that of "glue" between the MSI circuits. The result is an optimized system with more MSI packages than SSI packages, with 80 to 90% of the logic implemented with MSI. The applications in this book intend to prove this point by showing conventional and *unconventional* uses of MSI which make systems more economical and more reliable.

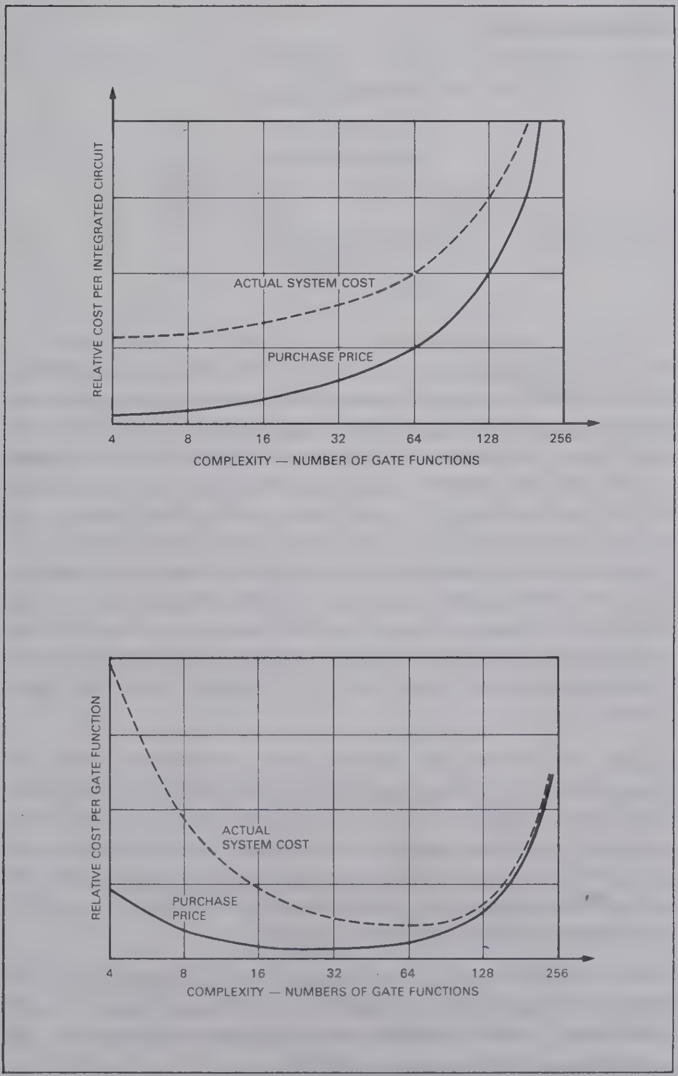
GENERAL SYSTEM DESIGN RULES

- Adapt system architecture to performance required and components used. Use *parallel* architecture and *fast* components for highest speed. Use *serial* architecture and *slow* components for slow systems, which reduces cost and power consumption. Use *parallel* architecture with *slow* components or *serial* architecture with *fast* components for intermediate speed requirements.
- Avoid asynchronous systems; convert them to synchronous. Synchronous systems are easier to design, debug and service. They are more reliable than asynchronous systems. A simple, inexpensive clock generator using less than one gate package may be sufficient to solve an inherently asynchronous problem in a synchronous manner.
- Use extra care with all clock signals to counters and registers and with trigger inputs to monostables. Avoid clock gating as much as possible; use the synchronous Enable inputs instead. Beware of the glitches on the outputs of decoders and similar combinatorial logic. Avoid slow rise times (>50 ns) and watch out for double pulses (overtones) from crystal oscillators. Most problems with inherently slow systems can be traced to double triggering of register and monostables due to poor clock and trigger signals. The designer of slow systems must be constantly aware of the fact that TTL components are capable of speeds of 10 to 50 MHz and that they react to trigger spikes invisible on an oscilloscope used for displaying slow events.
- Minimize the use of monostables and avoid RC elements in any signal path. Monostables are often used as a "quick and dirty" fix for an improperly designed system. Monostables are inherently linear circuits with limited noise immunity and this is a major disadvantage in noisy digital environments. A carefully designed synchronous system using edge triggered devices *rarely* needs a monostable.
- MSI designs should be based directly on system block diagrams. A gate-minimized logic design hides the basic system structure and a direct conversion to MSI is bound to be inefficient. It is always better to discard gate-minimized logic design and design with MSI directly from the original system block diagrams.
- Imaginatively explore MSI functional capabilities. The name applied to most MSI circuits merely describes the primary function of that device. A well-defined MSI device is far more versatile than the obvious function indicated by its name. A synchronous presettable counter can be used as a shift register, a decoder can be a data demultiplexer, and a multiplexer can be an efficient function generator. MSI devices are surprisingly versatile and this versatility should be used to advantage.

TTL/MSI DEVICE DESIGN CONSIDERATIONS

Thus far, this introduction has discussed TTL/MSI from the point of view of the system designer. However, it is also important for system designers to be aware of the design considerations faced by the TTL/MSI device designer. The design of TTL/MSI circuits is predominately influenced by three factors.

- COST — The device must be inexpensive to manufacture.
- POWER — The device must be able to dissipate its power while maintaining an acceptable junction temperature.
- PACKAGE — The device must be partitioned to perform a useful function with minimal input and output connections available in a practical package.



Cost

The chip and the package are the major influences on the manufacturing cost of TTL/MSI circuits. Chip cost does increase with complexity, but at low complexity this increase is less than proportional because chip connection pads actually dominate the chip area. With very high complexity, chip cost increases at a rate more than proportional to complexity because very complex and larger chips are subject to a lower manufacturing yield. Because the packaging cost for a given package is not affected by the chip complexity, it has a relatively larger impact on the cost of simple circuits. So, the cost of producing an integrated circuit rises slowly with low complexity chips, but then climbs steeply. Therefore, the cost per on-chip gate function has a minimum at a certain complexity. For random or near-random logic, this minimum cost is at the 50 to 100-gate complexity level; for regular memory arrays, it is approximately 256 bits. In addition, low manufacturing cost is only achieved when circuits are produced in volume. Volume production amortizes development costs and eliminates or minimizes manufacturing process problems earlier. The result is improved product availability at a lower cost.

Power

Power means heat. The power dissipation of a device in a 16-lead Dual In-line package should be kept below 500 mW to insure a tolerable junction temperature at the highest operating ambient temperature. Fast TTL circuits dissipate approximately 5 mW per internal gate which limits complexity to about 100 gates per package. Complexity beyond 100 gates requires either a larger package or slower, less power consuming technology such as Low Power TTL or CMOS.

Package

Another device design constraint is the number of leads available for inputs and outputs. The 16-lead Dual In-line Package has become the standard for TTL/MSI circuits. It is inexpensive, easy to use, and compact. Larger 24-lead packages are used, but are considerably more expensive, three times larger, heavier, and more difficult to handle. Moreover, many inputs and in particular outputs increase chip size, assembly cost, and power dissipation.

Thus, from the manufacturer's standpoint, the ideal TTL/MSI device meets the following criteria.

- complexity of 50 to 100 gates
- 16-lead DIP
- no more than 5 or 6 outputs
- 300 to 500 mW power dissipation
- application in many digital systems in different industries

INTRODUCTION

USER-ORIENTED TTL/MSI

With these considerations in mind, several semiconductor manufacturers have developed and are offering TTL/MSI families. The most successful are Texas Instrument's 7400 family and Fairchild's 9300 family. Both families are second-sourced (the original part # is retained) and copied by many manufacturers.

Fairchild 9300 TTL/MSI circuits were designed, developed and built with emphasis on user-oriented features.

- All available leads are used.
  When 50 or more gates function in a 16-lead package the user has no access to most internal nodes. Consequently, the circuit must be defined and partitioned very carefully, so that the resulting device can perform a useful function with the limited input/output connections. With some devices (a 1-of-10 decoder, for example), lead use is obvious. Others, however (serial shift registers), do not need all the leads for their basic function. In these cases, logic is added (input multiplexers, complementary outputs, clock control circuits) which uses all available leads and makes the circuit more versatile, often eliminating external gates.

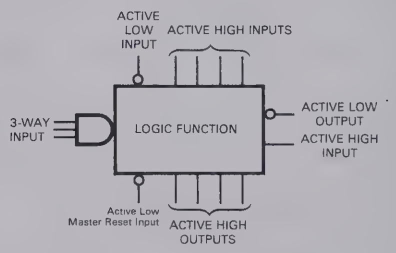
- Devices are self-extending.
  Most systems use MSI devices as parts of larger subsystems (four 4-bit counters form a 16-bit counter, four 8-input multiplexers form a 32-input multiplexer, five 5-bit comparators are used to build a 21-bit comparator). This expansion must be possible with minimal or no external gating. Otherwise, the cost, space, and power saving features of MSI are defeated. Enable inputs, decoded Terminal Count outputs, Group Signal outputs, etc., facilitate this expansion.

- Control and Clock polarities are compatible.
  Signal polarities (High or Low) throughout the TTL/MSI family have been selected so external inverters are not required for most applications. Decoder outputs are active Low, therefore Enable inputs are also active Low, because they are often driven by decoders or NAND gates. All Fairchild originated 9000 and 9300 flip-flops, registers, and counters change output states following the same clock transition (Low-to-High) avoiding clock skew problems encountered with mixed clock polarities.

- Inputs are buffered and encoded.
  Inputs to MSI devices, particularly control and clock inputs, often perfrom several internal functions. Without input buffers, this results in a fan in of several unit loads. The 9300 family has on-chip input buffers, avoiding this loading problem and offering several advantages. First, fan in is reduced, decreasing the number of SSI buffers required in a system. Second, on-chip buffers do not drastically increase the chip size, and because buffers are necessary somewhere in the system, designing them into the chip means they drive a known load at reduced voltage swing, saving power and gaining speed. SSI buffers, on the other hand, must be designed for worst-case supply voltage, noise margin, fan out, etc., and are therefore less efficient. Expanding the idea of on-chip input buffers leads to the placement of complete decoders on the chip inputs. Again, the additional chip area used is relatively small, but the advantages of increased function and/or less leads generally more than offsets the increase in chip size. Placing decoders on the chip also has important system consequences, substantially increasing the logic capability of the circuits, and resulting in greater logic flexibility.

- Sequential circuit timing is simplified.
  Almost all 9300 circuits operate synchronously. In a synchronous system, all output changes are actuated by a common clock transition, avoiding cascaded delays and decoding spikes which plague asynchronous systems. Edge triggered devices can accommodate combinatorial delays almost as long as a clock period cycle. They are less sensitive to noise and to variations in the clock duty cycle than ones catching Master/Slave circuits. This is discussed in detail under flip-flops (SSI) and in the Electrical Characteristics Chapter.

TERMINOLOGY

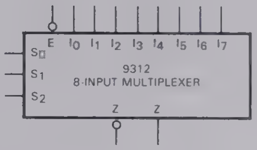
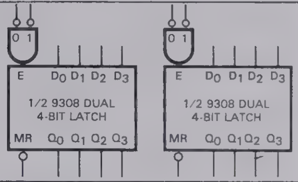
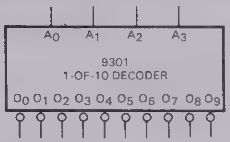
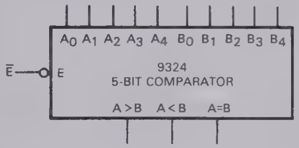
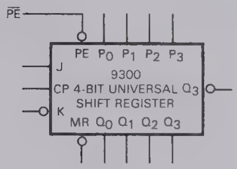
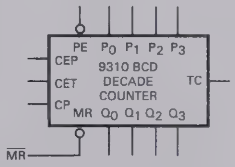
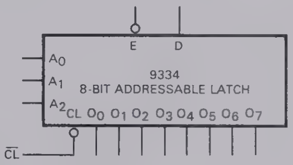
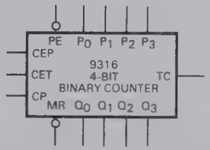
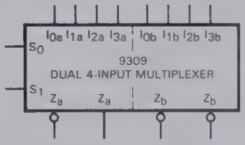
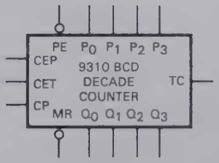


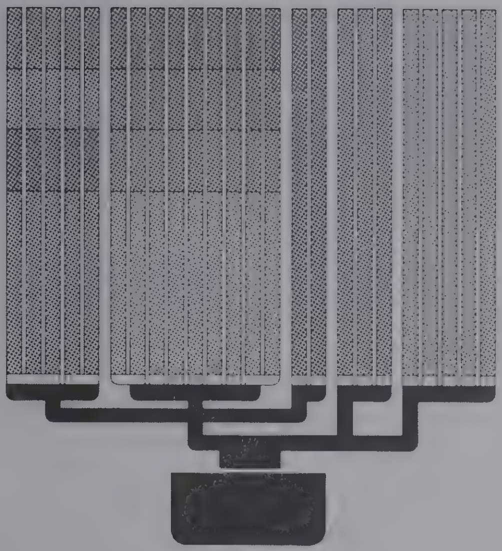
The logic symbols used to represent the MSI devices follow Mil Std 806B for logic symbols. MSI elements are represented by rectangular blocks with appropriate external AND/OR gates when necessary. A small circle at an external input means that the specific input is active Low; i.e., it produces the desired function, in conjunction with other inputs, if its voltage is the lower of the two logic levels in the system. A circle at the output indicates that when the function designated is True, the output is Low. Generally, inputs are at the top and left and outputs appear at the bottom and right of the logic symbol. An exception is the asynchronous Master Reset in some sequential circuits which is always at the left hand bottom corner.

Inputs and outputs are labeled with mnemonic letters as illustrated in the table opposite. Note that an active Low function labeled outside of the logic symbol is given a bar over the label, while the same function inside the symbol is labeled without the bar. When several inputs or outputs use the same letter, subscript numbers starting with zero are used in an order natural for device operation.

This nomenclature is used throughout this book and may differ from nomenclature used on data sheets (notably early 7400 MSI), where outputs use alphabetic subscripts or use number sequences starting with one.

INTRODUCTION

LABEL	MEANING	EXAMPLE
I_X	General term for inputs to combinatorial circuits.	 9312 8-INPUT MULTIPLEXER
J,K S,R D P	Inputs to JK, SR, and D flip-flops, latches, registers, and counters.	 1/2 9308 DUAL 4-BIT LATCH 1/2 9308 DUAL 4-BIT LATCH
A_X, S_X	Address or Select inputs, used to select an input, output, data route, junction, or memory location.	 9301 1-OF-10 DECODER
$\overline{E}$	Enable, active Low on all TTL/MSI.	 9324 5-BIT COMPARATOR
$\overline{PE}$	Parallel Enable, a control input used to synchronously load information in parallel into an otherwise autonomous circuit.	 9300 4-BIT UNIVERSAL SHIFT REGISTER
$\overline{MR}$	Master Reset, asynchronously resets all outputs to zero, overriding all other inputs.	 9310 BCD DECADE COUNTER
$\overline{CL}$	Clear, resets outputs to zero but does not override all other inputs.	 9334 8-BIT ADDRESSABLE LATCH
CP CE, CEP, CET	Clock Pulse, generally a High-to-Low-to-High transition. An active High clock (no circle) means outputs change on Low-to-High clock transition. Count Enable inputs for counters.	 9316 4-BIT BINARY COUNTER
Z_X, O_X, F_X	General terms for outputs of combinatorial circuits.	 9309 DUAL 4-INPUT MULTIPLEXER
Q_X TC	General term for outputs of sequential circuits. Terminal Count output (1111 for up binary counters, 1001 for up decimal counters, or 0000 for down counters).	 9310 BCD DECADE COUNTER

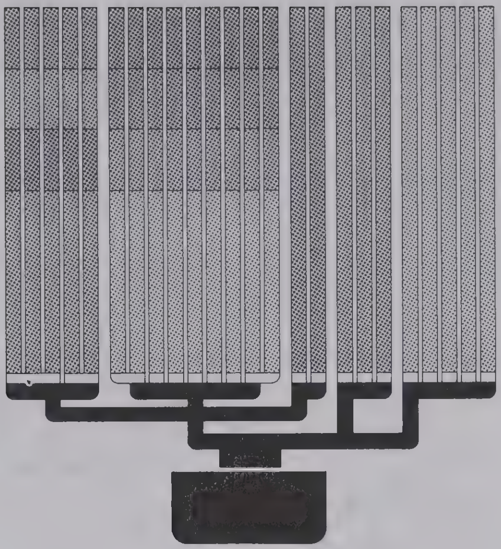


Multiplexers

Data Routing
Dual 10-Input Multiplexer
Left/Right Shift Parallel Load Register
Four Word Sorter
Serial Memory Applications
Data Commutation
Multiple Word Data Bussing
Data Transfer
Time Multiplexing and Data Routing
Memory Application
8-Step Control Sequence
Decade Multiplexer
Pattern Generator
15-Key to Binary Code Converter
Multiplexers as Function Generators
Adder, Full Subtractor
X of Y Pattern Detector

MULTIPLEXER SELECTION GUIDE

Function	Device	No. of Sections	No. of Inputs per Section	Enable Inputs	True Output	Complement Output	Select Delay, ns (typ)	Enable Delay, ns (typ)	Data Delay, ns (typ)	Power Dissipation mW (typ)	Fan Out	Leads
Quad 2-Input	9322, 74157	4	2	1	X		20	18	10	150	10.0 UL	16
	93L22	4	2	1	X		60	54	30	45	2.5 UL	16
	93S157, 74S157	4	2	1	X		10	8	5	250	12.5 UL	16
	93S257, 74S257	4	2	1	3-state		10	13	5	320	12.5 UL	16
	93S158, 74S158	4	2	1		X	8	7	4	195	12.5 UL	16
	93S258, 74S258	4	2	1		3-state	8	13	4	280	12.5 UL	16
Dual 4-Input	9309	2	4		X	X	15		10	150	10.0 UL	16
	93L09	2	4		X	X	45		30	38	2.5 UL	16
	93153	2	4	2	X		22	19	15	180	10.0 UL	16
	93S153, 74S153	2	4	2	X		12	10	6	225	12.5 UL	16
	93S253, 74S253	2	4	2	3-state		12	13	6	325	12.5 UL	16
8-Input	9312	1	8	1	X	X	18	15	10	135	10.0 UL	16
	93L12	1	8	1	X	X	54	45	30	36	2.5 UL	16
	93S12	1	8	1	X	X	12	10	7	190	12.5 UL	16
	9313	1	8	1	X	O.C.	25	22	18	135	10.0 UL	16
	93S13	1	8	1	3-state	3-state	12	10	7	215	12.5 UL	16
	93151, 74151	1	8	1	X	X	22	21	13	145	10.0 UL	16
	93S151, 74S151	1	8	1	X	X	17	10	8	225	12.5 UL	16
	93S251, 74S251	1	8	1	3-state	3-state	12	13	8	275	12.5 UL	16
	93152, 74152	1	8	1		X	22		13	130	10.0 UL	14
16-Input	93150, 74150	1	16	1		X	22	21	13	200	10.0 UL	24



MULTIPLEXERS

INTRODUCTION

Digital multiplexers are combinatorial (non-memory) devices controlled by a selector address which routes one of many input signals to the output. They can be considered semiconductor equivalents to multiposition switches or stepping switches.

Multiplexers are used for data routing and time division multiplexing. They can also generate complex logic functions. For example, the 9312 8-input/1-pole multiplexer generates any of the 2^{16} different logic functions of four variables; the 9309 4-input/2-pole multiplexer generates any two of the 256 functions of three variables; the 9322 2-input/4-pole multiplexer generates any four of the 16 functions of two variables. A single multiplexer package can replace several gate packages, saving printed circuit board area, interconnections, propagation delays, power dissipation, design effort, and component cost.

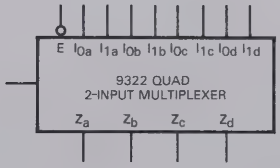
9322 QUAD 2-INPUT MULTIPLEXER

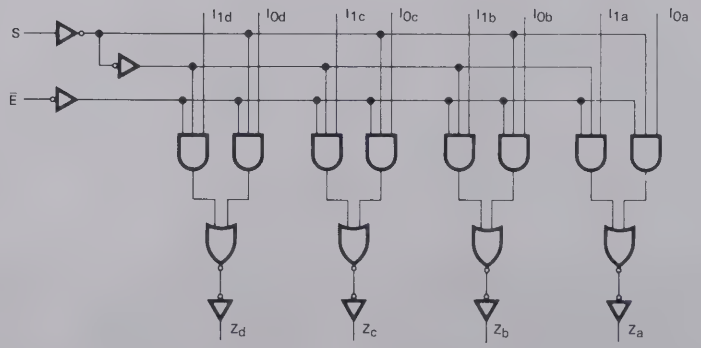
DESCRIPTION AND OPERATION

LEADS		LOADING
S	Common Select Input	1 UL
$\overline{E}$	Enable (Active Low) Input	1 UL
I_{0X}, I_{1X}	Multiplexer Inputs	1 UL
Z	Multiplexer Output	10 UL

CHARACTERISTICS		
TYPICAL DELAYS	I_0 to Z	10 ns
	$\overline{E}$ to Z	15 ns
	S to Z	19 ns
PACKAGE	16 Lead DIP or Flatpak	
TYPICAL POWER DISSIPATION	125 mW	

ENABLE	SELECT INPUT	INPUTS		OUTPUT
$\overline{E}$	S	I_{0X}	I_{1X}	Z_X
H	X	X	X	L
L	H	X	L	L
L	H	X	H	H
L	L	L	X	L
L	L	H	X	H

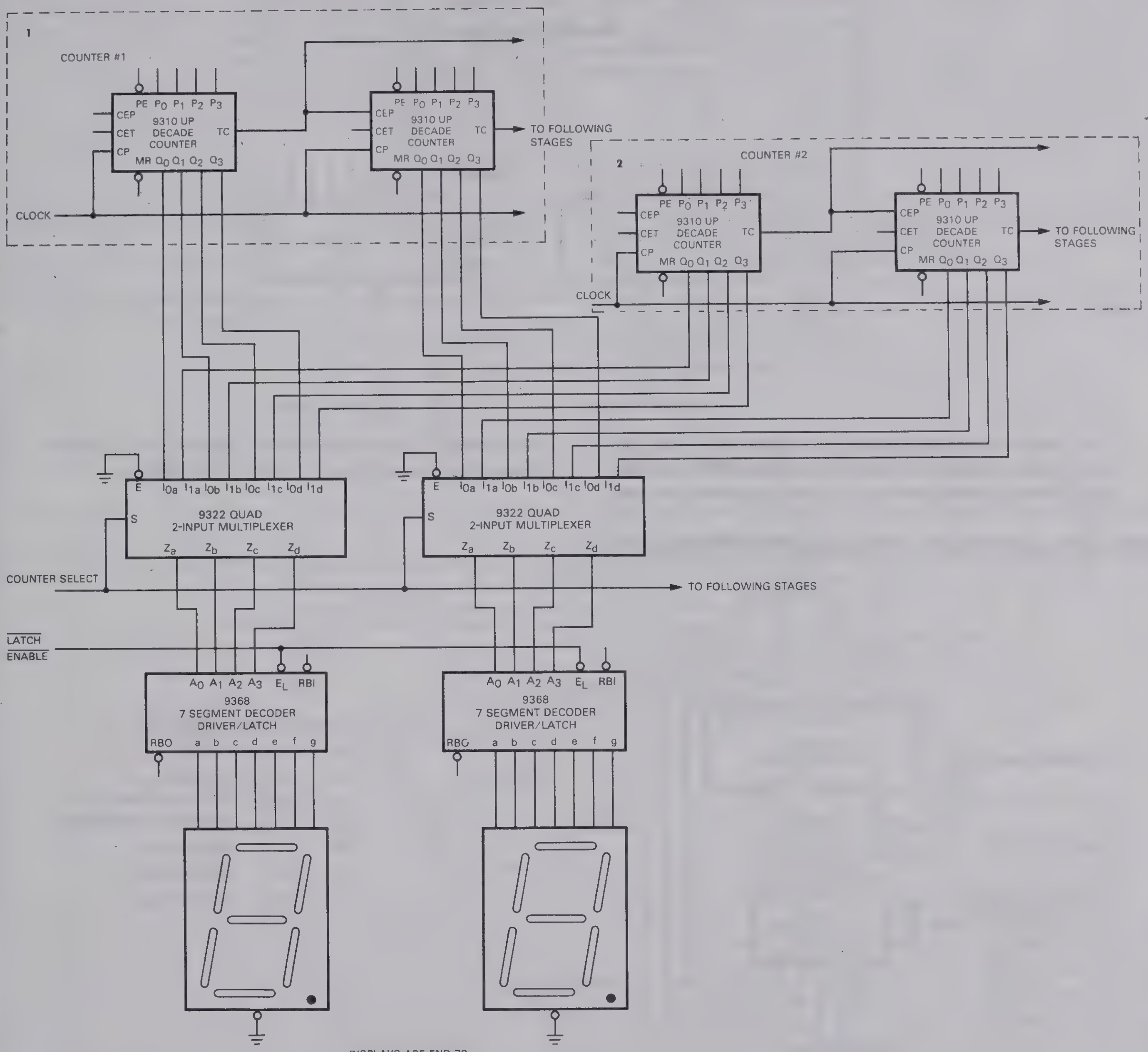

E
I_{0a}
I_{1a}
I_{0b}
I_{1b}
I_{0c}
I_{1c}
I_{0d}
I_{1d}
9322 QUAD
2-INPUT MULTIPLEXER
Z_a
Z_b
Z_c
Z_d


S
$\overline{E}$
I_{1d}
I_{0d}
I_{1c}
I_{0c}
I_{1b}
I_{0b}
I_{1a}
I_{0a}
Z_d
Z_c
Z_b
Z_a

The 9322 quad 2-input multiplexer has common input select logic, common active Low Enable and active High outputs. It allows four bits of data to be switched in parallel to the appropriate outputs from four 2-bit data sources. When the Enable is not active, all the outputs are held Low.

9322 QUAD 2-INPUT MULTIPLEXER APPLICATIONS

DATA ROUTING


1
COUNTER #1
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
TO FOLLOWING STAGES
CLOCK
2
COUNTER #2
E
S
I0a I1a I0b I1b I0c I1c I0d I1d
9322 QUAD 2-INPUT MULTIPLEXER
Za Zb Zc Zd
COUNTER SELECT
TO FOLLOWING STAGES
LATCH ENABLE
A0 A1 A2 A3 EL RBI
9368 7 SEGMENT DECODER DRIVER/LATCH
RBO a b c d e f g

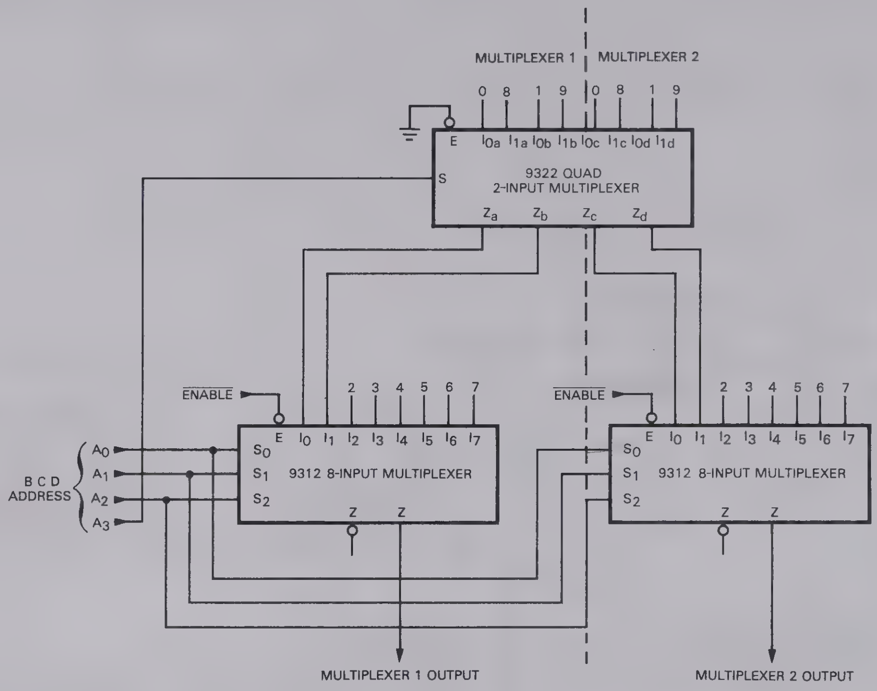
DISPLAYS ARE FND 70

Multiplexers usually route data from one of several sources to one destination. One typical application is shown here. This system displays the contents of one of two multidigit BCD counter banks. The 9322 multiplexers select one of the two counters; when the Counter Select line is Low, counter 1 is selected, when it is High, counter 2 is selected. The multiplexer outputs feed into the 9368 BCD to 7-segment decoder drivers with input latches.

The display follows the selected counter when the Latch Enable input is Low. When this line is High, the display is no longer affected by input changes, but retains the information that was applied prior to the Low-to-High transition of the Latch Enable. The 9368 interfaces directly with common cathode LED displays such as the FND 70. For driving incandescent displays with this circuit, simply replace the 9368 devices with 9370 7-segment decoder/driver/latches.

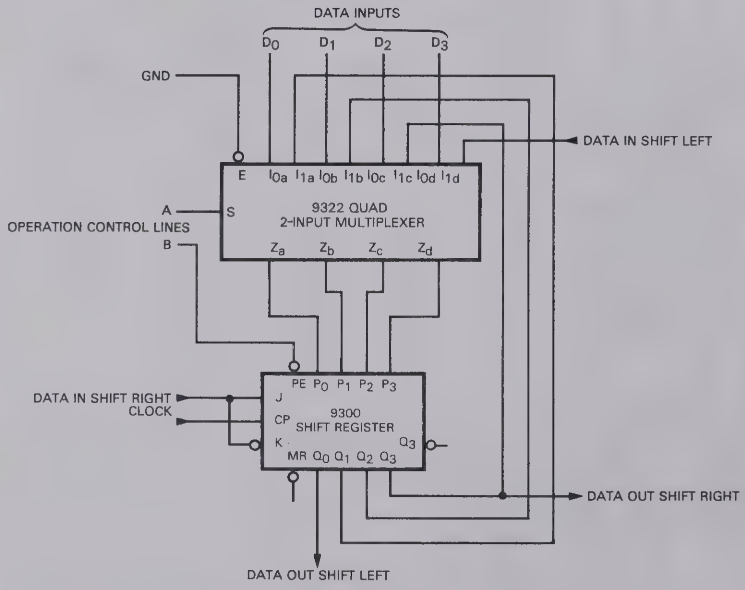
9322 QUAD 2-INPUT MULTIPLEXER APPLICATIONS

DUAL 10-INPUT MULTIPLEXER


MULTIPLEXER 1
MULTIPLEXER 2
0 8 1 9
0 8 1 9
E
I0a I1a I0b I1b I0c I1c I0d I1d
S
9322 QUAD
2-INPUT MULTIPLEXER
Za Zb Zc Zd
ENABLE
2 3 4 5 6 7
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
ENABLE
2 3 4 5 6 7
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
BCD ADDRESS
A0
A1
A2
A3
MULTIPLEXER 1 OUTPUT
MULTIPLEXER 2 OUTPUT

This data routing application is a part of a dual 10-input, BCD addressed multiplexer. One 9322 and two 9312 8-input multiplexers are used to route two sets of 10 inputs to two output lines. Other decade multiplexing circuits are shown later in this section.

SHIFT LEFT, SHIFT RIGHT PARALLEL LOAD REGISTER


DATA INPUTS
D0 D1 D2 D3
GND
DATA IN SHIFT LEFT
E
I0a I1a I0b I1b I0c I1c I0d I1d
9322 QUAD
2-INPUT MULTIPLEXER
S
A
OPERATION CONTROL LINES
B
Za Zb Zc Zd
PE P0 P1 P2 P3
J
DATA IN SHIFT RIGHT
CLOCK
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
DATA OUT SHIFT RIGHT
DATA OUT SHIFT LEFT

OPERATION CODE LIST

A	B	Operation
L	L	Parallel Load Data
H	L	Shift Left
L	H	Shift Right
H	H	Shift Right

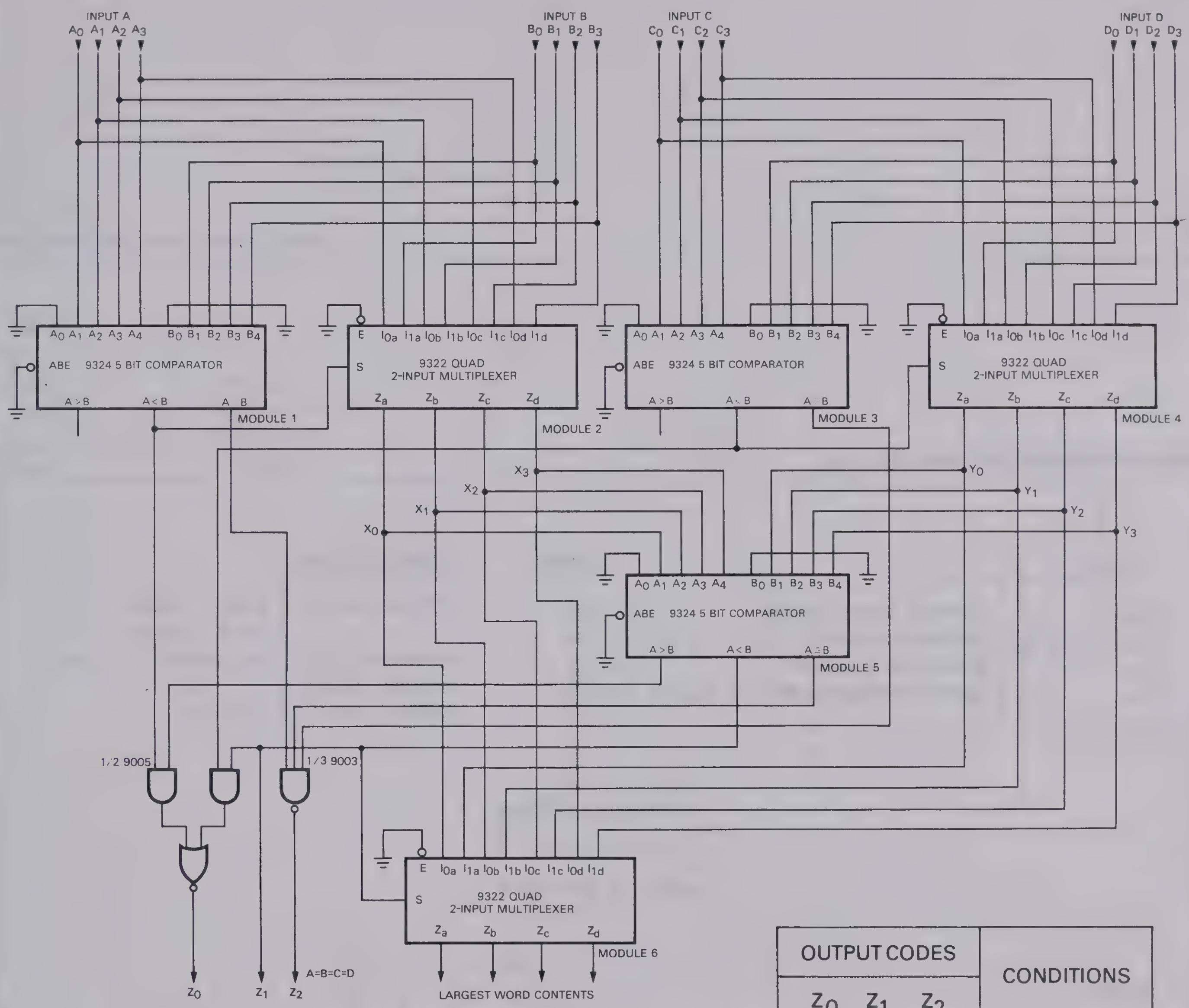
The circuit shown above is a shift left, shift right, parallel load register. The 9300 shift register right shifts synchronously with the Low-to-High clock transitions when the $\overline{PE}$ input is High. When $\overline{PE}$ is Low, the rising clock enters data on the P_0 through P_3 inputs to all four stages of the shift register.

The multiplexer feeds into each P input either new data or the contents of the stage to the right. This latter mode is the shift left capability. Data on the Q_n output is multiplexed to be fed to the P_{n-1} input. On the next clock pulse, the data is loaded into the n-1 stage and appears at Q_{n-1}, shifted one stage to the left. The 9322 shifts the data or enters new data at the discretion of the process controller. Connections to the preceding and following stages of a larger register are shown.

The use of 9322s to provide shifting capability is advantageous in any unit which has parallel inputs and storage. The 9310 decade counter, the 9316 binary counter, the 9338 multiple-port register are all good candidates for this procedure which allows them to both shift and load.

9322 QUAD 2-INPUT MULTIPLEXER APPLICATIONS

WORD SORTER FOR COMPARING FOUR 4-BIT WORDS


INPUT A
A0 A1 A2 A3
INPUT B
B0 B1 B2 B3
INPUT C
C0 C1 C2 C3
INPUT D
D0 D1 D2 D3
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
ABE 9324 5 BIT COMPARATOR
A>B A<B A=B
MODULE 1
E I0a I1a I0b I1b I0c I1c I0d I1d
S 9322 QUAD 2-INPUT MULTIPLEXER
Za Zb Zc Zd
MODULE 2
MODULE 3
MODULE 4
X0 X1 X2 X3
Y0 Y1 Y2 Y3
MODULE 5
1/2 9005
1/3 9003
MODULE 6
Z0 Z1 Z2
A=B=C=D
LARGEST WORD CONTENTS

OUTPUT CODES			CONDITIONS
Z_0	Z_1	Z_2	
H	L	H	A≥B,C,D
L	L	H	B≥A,C,D
H	H	H	C≥A,B,D
L	H	H	D≥A,B,C
H	L	L	A=B=C=D

This circuit uses 9322 and 9324 5-bit comparators to compare and order four 4-bit words. The four words are applied as shown. The result of the comparison of A and B in module 1 causes the larger of the two to appear at the outputs of module 2. Likewise, the larger of C and D appears at the outputs of module 4 as a result of the decision of module 3. Module 5 compares these two larger words and causes the larger of these to appear at the output of module 6. Lines Z_0 and Z_1 indicate which of the four words appear. If all of the words are identical, Z_2 is Low.

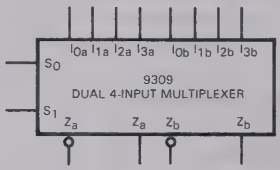
The operation is completely asynchronous. When the Control and Data outputs have settled, the largest of the four words appears at the Z_a – Z_d outputs (Module 6) and can be stored in a memory or register. If the appropriate input word (as indicated by the Z_0 and Z_1 outputs) is forced to zero, the next largest word appears at the Z_a – Z_d outputs. This operation can be repeated four times, after which the input words have been sorted in descending order. The circuit can be expanded to more words and/or more bits by adding 9324 comparators and appropriate multiplexers.

9309 DUAL 4-INPUT MULTIPLEXER

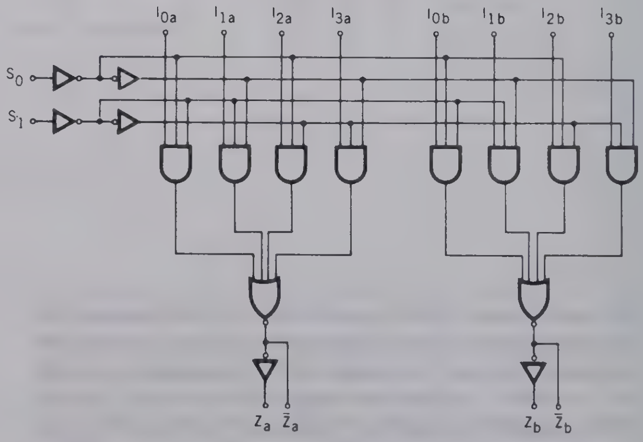
DESCRIPTION AND OPERATION

LEADS		LOADING
S_0, S_1	Common Select Inputs	1 UL
$I_{0X}, I_{1X}, I_{2X}, I_{3X}$	Multiplexer Inputs	1 UL
Z_X	Multiplexer Outputs	10 UL
$\overline{Z}_X$	Complementary Multiplexer Outputs	9 UL

CHARACTERISTICS	
TYPICAL DELAYS	S to Z 24 ns I to $\overline{Z}$ 9 ns
PACKAGE	16 Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	150 mW


I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
9309
DUAL 4-INPUT MULTIPLEXER
S_1
Z_a Z_a Z_b Z_b

SELECT INPUTS		INPUTS				OUTPUTS	
S_0	S_1	I_{0X}	I_{1X}	I_{2X}	I_{3X}	Z_X	$\overline{Z}_X$
L	L	L	X	X	X	L	H
L	L	H	X	X	X	H	L
H	L	X	L	X	X	L	H
H	L	X	H	X	X	H	L
L	H	X	X	L	X	L	H
L	H	X	X	H	X	H	L
H	H	X	X	X	L	L	H
H	H	X	X	X	H	H	L

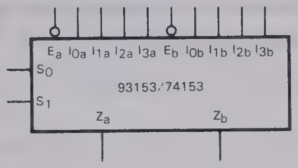

I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
S_1
Z_a $\overline{Z}_a$
Z_b $\overline{Z}_b$

The 9309 is two 4-input multiplexers with common select logic and True and Complement outputs. The 2-bit code on the S inputs selects one of the four inputs in each of the two sections and routes it to the appropriate output. In addition to conventional multiplexing, the 9309 generates any two of the 256 possible functions of three variables, a useful feature for implementing random control functions.

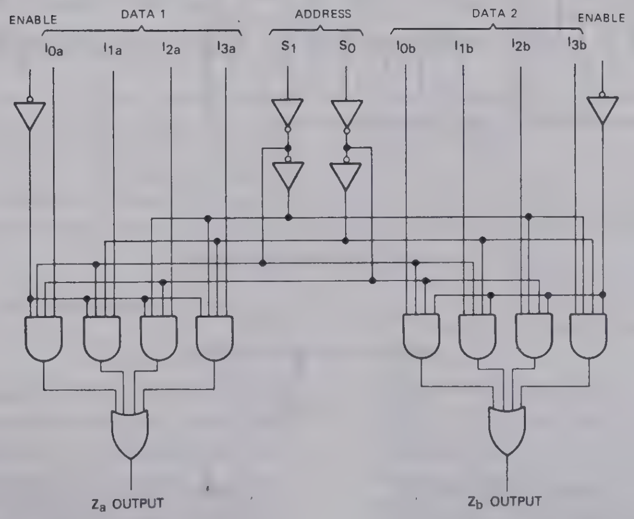
93153/74153 DUAL 4-INPUT MULTIPLEXER

DESCRIPTION AND OPERATION

LEADS		LOADING
$\overline{E}_a, \overline{E}_b$	Enable	1 UL
S_0, S_1	Address Inputs	1 UL
I_{0a}-$_{3a}$, I_{0b}-$_{3b}$	Data Inputs	1 UL
Z_a, Z_b	Outputs	10 UL


$\overline{E}_a$
I_{0a}
I_{1a}
I_{2a}
I_{3a}
$\overline{E}_b$
I_{0b}
I_{1b}
I_{2b}
I_{3b}
S_0
S_1
93153/74153
Z_a
Z_b

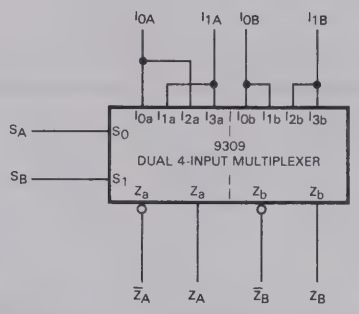
ADDRESS INPUTS		DATA INPUTS				ENABLE	OUTPUT
S_1	S_0	I_{0X}	I_{1X}	I_{2X}	I_{3X}	$\overline{E}_X$	Z_X
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
L	H	X	L	X	X	L	L
L	H	X	H	X	X	L	H
H	L	X	X	L	X	L	L
H	L	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H


ENABLE
DATA 1
I_{0a}
I_{1a}
I_{2a}
I_{3a}
ADDRESS
S_1
S_0
DATA 2
I_{0b}
I_{1b}
I_{2b}
I_{3b}
ENABLE
Z_a OUTPUT
Z_b OUTPUT

The 93153/74153 is a dual 4-input multiplexer with common Select inputs and separate Enable inputs. It allows two bits of data to be selected from two sets of 4-input sources.

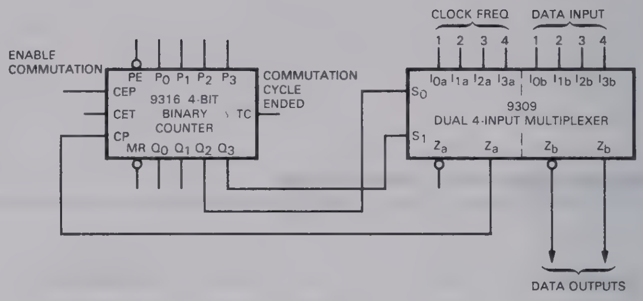
9309 DUAL 4-INPUT MULTIPLEXER APPLICATIONS

DUAL 2-INPUT MULTIPLEXER


I_{0A}
I_{1A}
I_{0B}
I_{1B}
S_A
S_B
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
S_1
9309
DUAL 4-INPUT MULTIPLEXER
$\overline{Z}_a$ Z_a $\overline{Z}_b$ Z_b
$\overline{Z}_A$
Z_A
$\overline{Z}_B$
Z_B

The 9309 dual 4-input multiplexer can be used as two independent 2-input multiplexers by connecting the inputs in parallel as shown above.

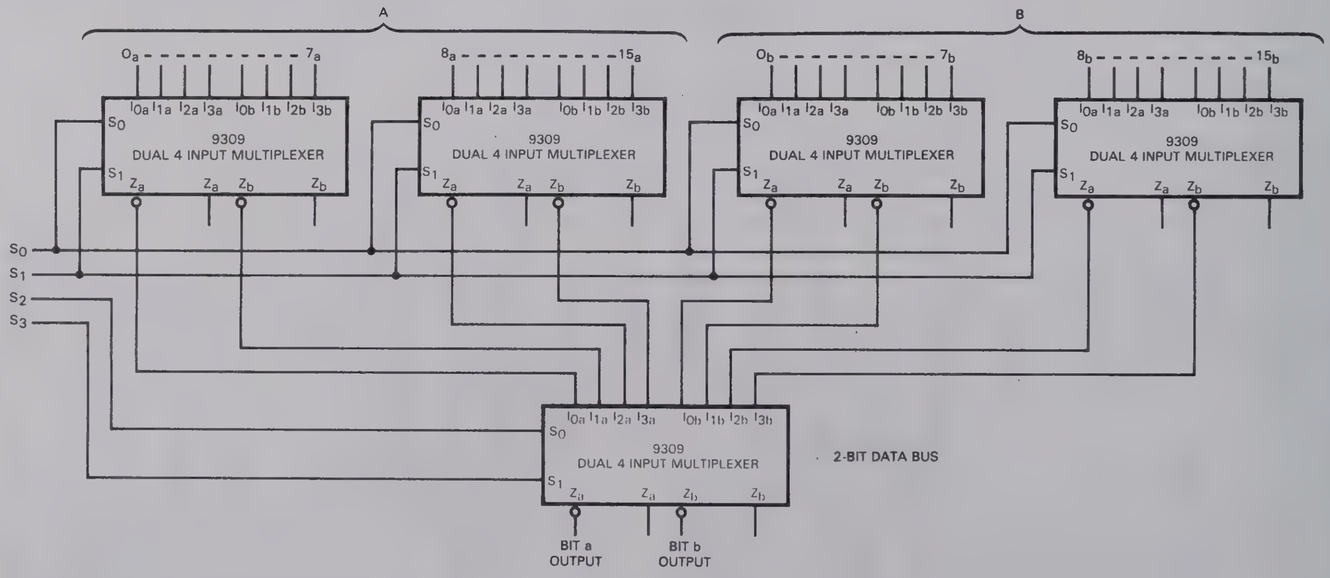
DATA COMMUTATION


CLOCK FREQ
DATA INPUT
1 2 3 4
1 2 3 4
ENABLE COMMUTATION
PE
P_0 P_1 P_2 P_3
CEP
CET
CP
MR
Q_0 Q_1 Q_2 Q_3
9316 4-BIT BINARY COUNTER
TC
COMMUTATION CYCLE ENDED
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
S_1
9309
DUAL 4-INPUT MULTIPLEXER
$\overline{Z}_a$ Z_a $\overline{Z}_b$ Z_b
DATA OUTPUTS

THE DATA COMMUTATOR SPENDS A VARIABLE AMOUNT OF TIME IN EACH POSITION. CLOCK FREQUENCY DETERMINES AMOUNT OF TIME SAMPLING DATA INPUT 'X'. THE TIME SPENT IN EACH POSITION IS RECIPROCALLY RELATED TO FREQUENCY INPUT.

This circuit uses a 9309 dual 4-input multiplexer and a 9316 up binary counter for data commutation. In each selected position, $I_0 - I_3$, the data commutator samples data for a length of time that is determined by the clock frequencies at the four I_b multiplexer inputs. The 9316 CEP input serves as a commutation enable, while the 9316 TC output indicates the end of the commutation cycle.

MULTIPLE WORD DATA BUSSING


A
B
0_a - - - - - - - - - - - 7_a
8_a - - - - - - - - - - - 15_a
0_b - - - - - - - - - - - 7_b
8_b - - - - - - - - - - - 15_b
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
S_1
9309
DUAL 4 INPUT MULTIPLEXER
$\overline{Z}_a$ Z_a $\overline{Z}_b$ Z_b
S_0
S_1
S_2
S_3
2-BIT DATA BUS
BIT a OUTPUT
BIT b OUTPUT

Five 9309 dual 4-bit multiplexers connected as shown can be used to switch two bits of data from one of 16 words to a 2-bit data bus. The address supplied to the S_0, S_1, S_2, S_3 inputs selects the word to be transferred. If 12-bit words are to be transferred to a 12-bit bus, the circuit is repeated six times. The complementary outputs are used at both levels in order to minimize the through delay. (The Z output is derived from the $\overline{Z}$ output through an additional inverter and is therefore delayed by one additional gate delay.) The two inversions of the two multiplexer levels cancel, so that data is not inverted.

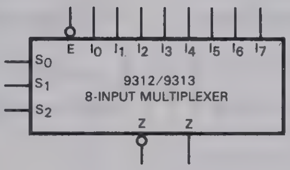
9312/9313 8-INPUT MULTIPLEXERS

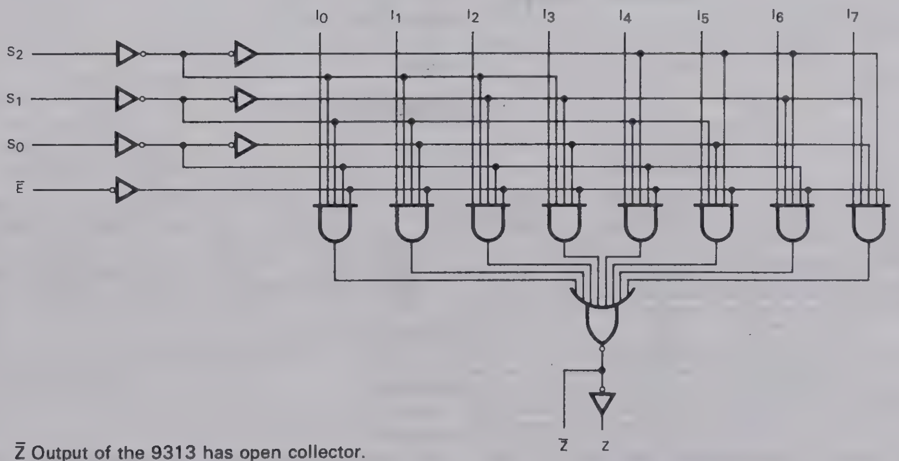
DESCRIPTION AND OPERATION

LEADS		LOADING
S_0, S_1, S_2	Select Inputs	1 UL
$\overline{E}$	Enable (Active Low) Input	1 UL
I_0 to I_7	Multiplexer Inputs	1 UL
Z	Multiplexer Output	10 UL
$\overline{Z}$	Complementary Multiplexer Output	9 UL

CHARACTERISTICS	
TYPICAL DELAYS	S to Z 24 ns $\overline{E}$ to $\overline{Z}$ 14 ns I to $\overline{Z}$ 9 ns
PACKAGE	16 Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	135 mW

INPUTS												OUTPUTS	
$\overline{E}$	S_2	S_1	S_0	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	$\overline{Z}$	Z
H	X	X	X	X	X	X	X	X	X	X	X	H	L
L	L	L	L	L	X	X	X	X	X	X	X	H	L
L	L	L	L	H	X	X	X	X	X	X	X	L	H
L	L	L	H	X	L	X	X	X	X	X	X	H	L
L	L	L	H	X	H	X	X	X	X	X	X	L	H
L	L	H	L	X	X	L	X	X	X	X	X	H	L
L	L	H	L	X	X	H	X	X	X	X	X	L	H
L	L	H	H	X	X	X	L	X	X	X	X	H	L
L	L	H	H	X	X	X	H	X	X	X	X	L	H
L	H	L	L	X	X	X	X	L	X	X	X	H	L
L	H	L	L	X	X	X	X	H	X	X	X	L	H
L	H	L	H	X	X	X	X	X	L	X	X	H	L
L	H	L	H	X	X	X	X	X	H	X	X	L	H
L	H	H	L	X	X	X	X	X	X	L	X	H	L
L	H	H	L	X	X	X	X	X	X	H	X	L	H
L	H	H	H	X	X	X	X	X	X	X	L	H	L
L	H	H	H	X	X	X	X	X	X	X	H	L	H





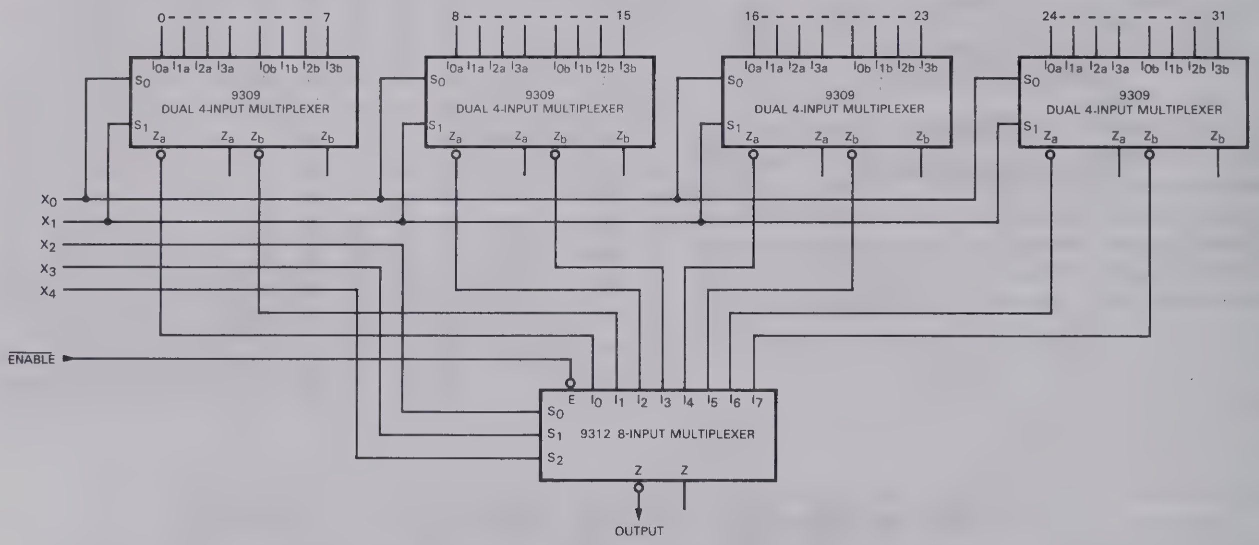
$\overline{Z}$ Output of the 9313 has open collector.

The 9312 and 9313 are 8-input multiplexers which select one bit of data from up to eight sources. They have internal select decoding, active Low Enable and Complementary outputs. When the Enable input is active, (Low) data is routed from one particular multiplexer input to the outputs according to the 3-bit code applied to the Select inputs. When the Enable is inactive, (High) the Z output is Low and the $\overline{Z}$ output is High regardless of all other input conditions.

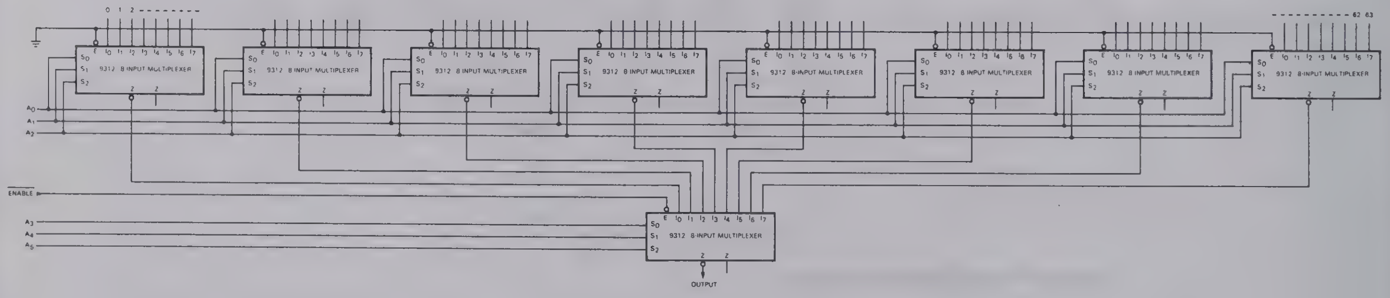
On the 9312, both the Z and $\overline{Z}$ outputs have a conventional TTL active pull up structure. On the 9313, the $\overline{Z}$ output has an open collector which simplifies multiplexer expansion by Or-tying the $\overline{Z}$ outputs of several 9313s. Since the Z output is generated by inverting the $\overline{Z}$ output, it is available as a TTL output even from an array of OR-tied 9313s.

9312 8-INPUT MULTIPLEXER APPLICATIONS

TIME MULTIPLEXING OR DATA ROUTING


0 - - - 7
8 - - - 15
16 - - - 23
24 - - - 31
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309
DUAL 4-INPUT MULTIPLEXER
Za Za Zb Zb
X0
X1
X2
X3
X4
ENABLE
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
OUTPUT

a. 32-INPUT MULTIPLEXING CIRCUIT

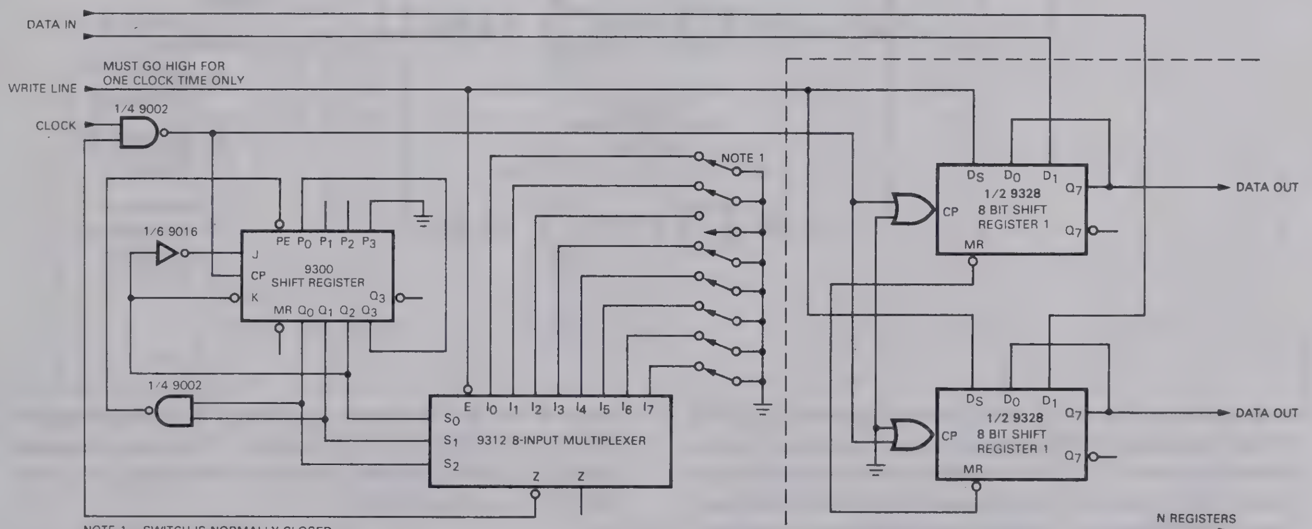

0 1 2 - - -
- - - 62 63
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
A0
A1
A2
ENABLE
A3
A4
A5
OUTPUT

b. 64-INPUT MULTIPLEXING CIRCUIT

Alone, the 9309 and the 9312 permit time multiplexing of a maximum of four and eight data lines, respectively. By cascading these devices in two or more levels, the number of inputs can be increased. The circuit in *a* above shows two levels of multiplexers cascaded to implement a 32-input multiplexer with a delay of about 50 ns. It can be expanded to the 64-input multiplexer shown in *b* without adding delay. In the 32-input multiplexer, the 9312 Enable can be used to gate the selected data out. Note that the negative outputs are used at both levels to improve through delay. As indicated in the circuit schematic, the assertion output is generated by re-inverting the negative output and is therefore slower.

9312 MULTIPLEXER APPLICATIONS

MEMORY


DATA IN
MUST GO HIGH FOR ONE CLOCK TIME ONLY
WRITE LINE
CLOCK
1/4 9002
1/6 9016
PE P0 P1 P2 P3
J
CP
K
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
1/4 9002
NOTE 1
S0
S1
S2
E I0 I1 I2 I3 I4 I5 I6 I7
9312 8-INPUT MULTIPLEXER
Z Z
DS D0 D1
1/2 9328
8 BIT SHIFT REGISTER 1
Q7
CP
MR
DATA OUT
DS D0 D1
1/2 9328
8 BIT SHIFT REGISTER 1
Q7
CP
MR
DATA OUT
N REGISTERS

NOTE 1 SWITCH IS NORMALLY CLOSED
PUSH BUTTON BAILING SWITCH
OPEN BEFORE MAKE

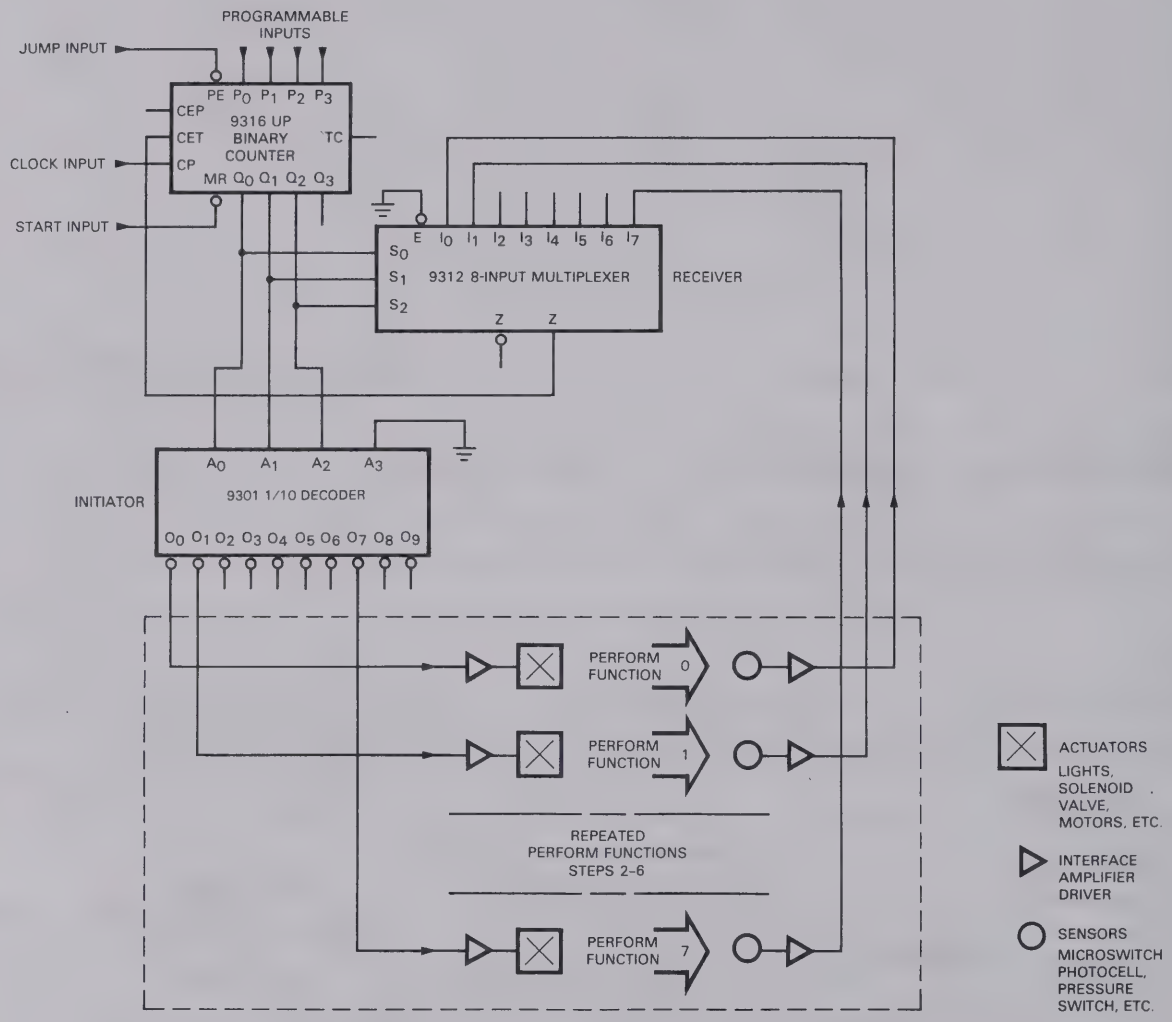
A 9328 dual 8-bit shift register in conjunction with a 9300 universal 4-bit shift register counter and a 9312 8-input multiplexer, offers an 8-word-by-n-bit memory with parallel input and parallel output capabilities.

A switch with eight push buttons is used to select the data word for the data output. The push button for this output, when depressed, allows enough clock pulses to the counter and shift registers to bring the selected word to the last position in the registers, thus providing that word as an output. In effect, the 9300 counter scans the 8-input multiplexer until a High is detected on a 9312 input (an open switch) which causes gate G1 to block the system clock.

To write into any location, that location is selected by a push button, the desired data is furnished to the 9328 inputs and the write line is held High for one clock period. The High write line both releases the counter, shifts registers, and at the same time selects the parallel data in (in place of the recirculation output) as an input to the shift register. Expansion of this memory to any even number of bits is limited only by the drive capability of the write input and the clock.

9312 8-INPUT MULTIPLEXER APPLICATIONS

8-STEP CONTROL SEQUENCE



PROGRAMMABLE INPUTS
JUMP INPUT
CLOCK INPUT
START INPUT
PE
P_0 P_1 P_2 P_3
CEP
CET
CP
MR
9316 UP BINARY COUNTER
TC
Q_0 Q_1 Q_2 Q_3
E
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
S_0
S_1
S_2
9312 8-INPUT MULTIPLEXER
RECEIVER
Z
Z
A_0 A_1 A_2 A_3
INITIATOR
9301 1/10 DECODER
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7 O_8 O_9
PERFORM FUNCTION 0
PERFORM FUNCTION 1
REPEATED PERFORM FUNCTIONS STEPS 2-6
PERFORM FUNCTION 7
ACTUATORS
LIGHTS, SOLENOID VALVE, MOTORS, ETC.
INTERFACE AMPLIFIER DRIVER
SENSORS
MICROSWITCH PHOTOCELL, PRESSURE SWITCH, ETC.

In this circuit, the 9312 multiplexer is used as part of a control sequencer which steps through eight steps, each initiating a test. The controller advances only when the test result is positive. It can also cycle through any portion of the sequence or jump, conditionally or unconditionally, to any other step.

The 9316 counter is the modulo 8 program counter; its state is decoded by the O_0 – O_7 outputs of the 9301 which initiate the program steps. The counter also addresses the 9312 multiplexer which acts as a receiver. When the 9312 input is Low, the test indicates that the program step is not yet completed. When the input is High, the test indicates completion of the program step.

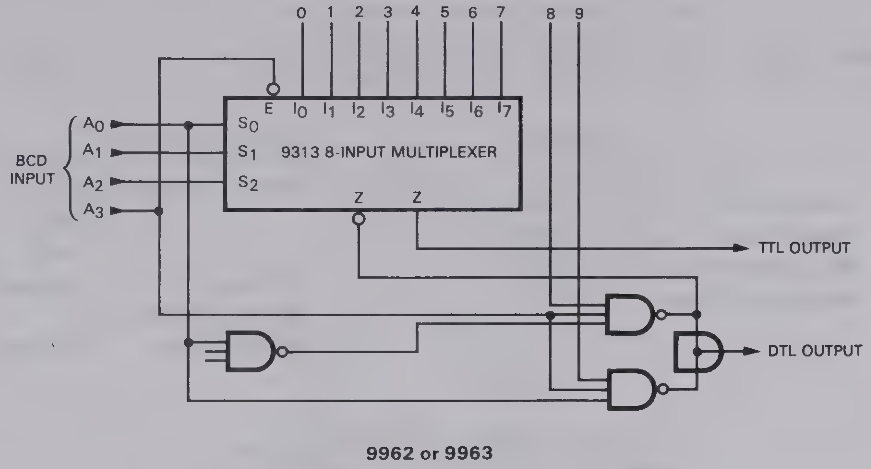
The active Low Start pulse initiates the sequence by resetting the counter. This state is decoded and activates the 9301 O_0 (step zero). This step lasts until the 9312 I_0 is activated which in turn activates the count Enable input of the 9316 counter, causing it to advance on the next Low-to-High clock transition. The next decoder output is then activated and this state lasts until the 9312 I_1 is activated, etc.

When the multiplexer inputs are tied High permanently, the sequencer advances to the next state on each subsequent clock pulse. Conditional or unconditional jumps can be made by activating the counter $\overline{PE}$ input with one of the decoder outputs and feeding the destination address into the counter's programmable inputs.

The system can be made asynchronous by using the $\overline{Z}$ output of the multiplexer as the clock source for the counter, which then advances whenever the multiplexer input goes from High to Low. This system is easily expanded to 16 or more control steps.

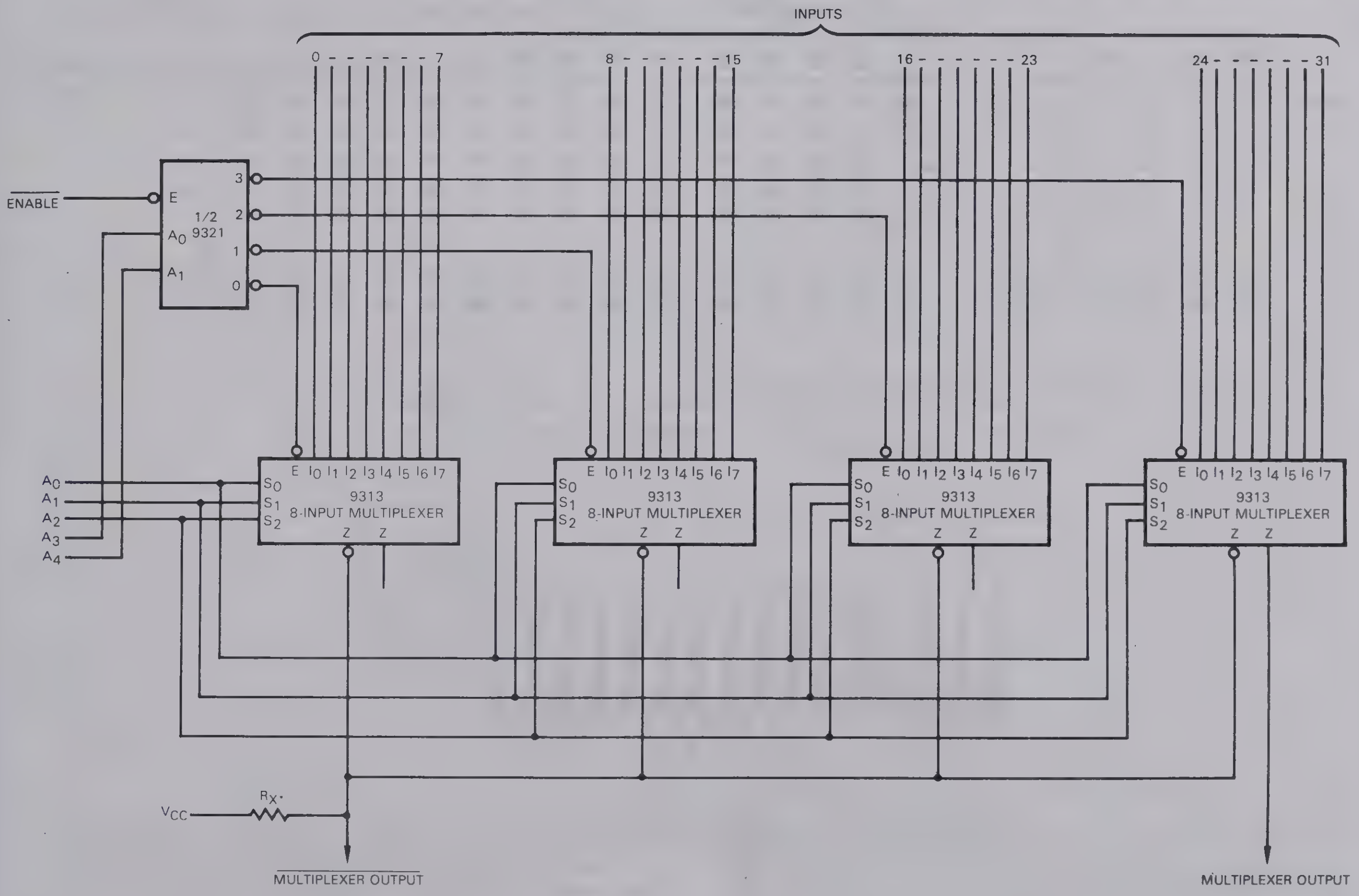
9313 8-INPUT MULTIPLEXER APPLICATIONS

DECADE MULTIPLEXER


0 1 2 3 4 5 6 7
8 9
BCD INPUT
A_0
A_1
A_2
A_3
E
S_0
S_1
S_2
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
9313 8-INPUT MULTIPLEXER
$\overline{Z}$ Z
TTL OUTPUT
DTL OUTPUT
9962 or 9963

This circuit provides a decade multiplexer with complementary outputs and uses only two packages; a 9313 and a triple 3-input gate. This circuit makes use of the OR-tie capabilities of the 9313 $\overline{Z}$ output and the DTL gates.

DIGITAL 32-INPUT MULTIPLEXER


INPUTS
0 - - - - - - 7
8 - - - - - - 15
16 - - - - - - 23
24 - - - - - - 31
$\overline{ENABLE}$
E
1/2 9321
A_0
A_1
3
2
1
0
A_0
A_1
A_2
A_3
A_4
E I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
S_0
S_1
S_2
9313 8-INPUT MULTIPLEXER
$\overline{Z}$ Z
V_{CC}
R_X
$\overline{\text{MULTIPLEXER OUTPUT}}$
MULTIPLEXER OUTPUT

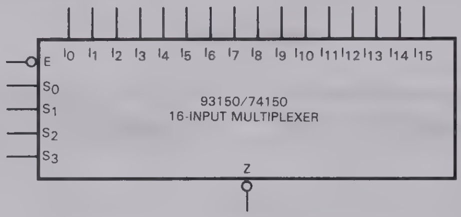
This 32-input digital multiplexer uses the 9313 open collector device with the $\overline{Z}$ outputs OR-tied. Up to seven 9313s can be wired-OR using a pull up resistor (2kΩ), and the $\overline{Z}$ output to drive one unit load. The True output may be taken from any of the 9313 Z outputs.

93150/74150 16-INPUT MULTIPLEXER

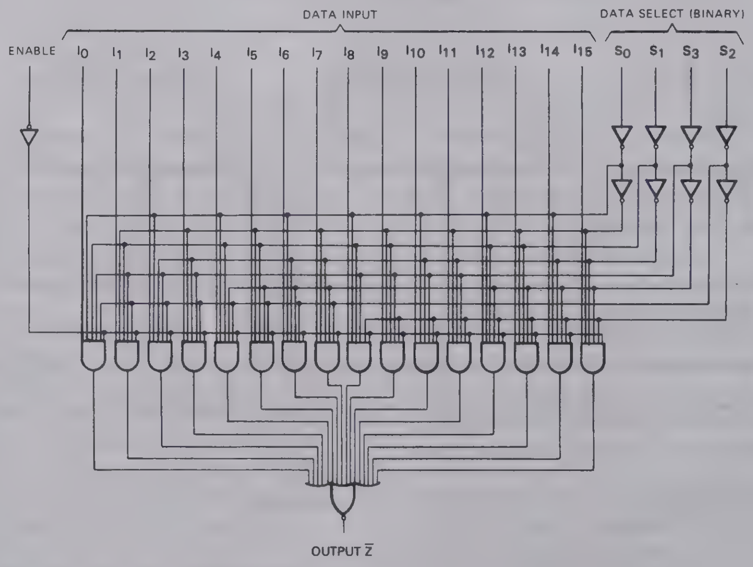
DESCRIPTION AND OPERATION

LEADS		LOADING
S_0, S_1, S_2, S_3	Select Input	1 UL
$\overline{E}$	Enable Input	1 UL
I_0 through I_{15}	Multiplexer Inputs	1 UL
$\overline{Z}$	Multiplexer Output	10 UL

CHARACTERISTICS		
TYPICAL DELAYS	S to $\overline{Z}$	23 ns
	$\overline{E}$ to $\overline{Z}$	17 ns
	I to $\overline{Z}$	10 ns
PACKAGE	24 Lead DIP or Flatpak	
TYPICAL POWER DISSIPATION	200 mW	


E
I0 I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13 I14 I15
S0
S1
S2
S3
93150/74150
16-INPUT MULTIPLEXER
Z

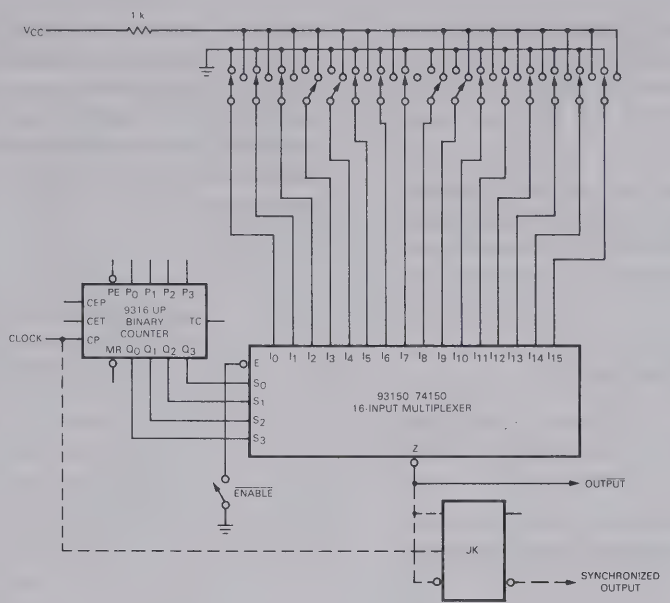
INPUTS																					OUTPUT
S_3	S_2	S_1	S_0	$\overline{E}$	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	$\overline{Z}$
X	X	X	X	H	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	H
L	L	L	L	L	L	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	H
L	L	L	L	L	H	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	L
L	L	L	H	L	X	L	X	X	X	X	X	X	X	X	X	X	X	X	X	X	H
L	L	L	H	L	X	H	X	X	X	X	X	X	X	X	X	X	X	X	X	X	L
			≈		≈															≈	
H	H	H	H	L	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	L	H
H	H	H	H	L	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	H	L


DATA INPUT
DATA SELECT (BINARY)
ENABLE
I0 I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13 I14 I15
S0 S1 S3 S2
OUTPUT Z̄

The 93150/74150 16-input multiplexer can select one bit of data from 16 sources. This data is routed according to a 4-input binary code applied to the Select inputs and appears inverted at the output. When the active Low Enable is inactive, the multiplexer output is High, regardless of all other input conditions.

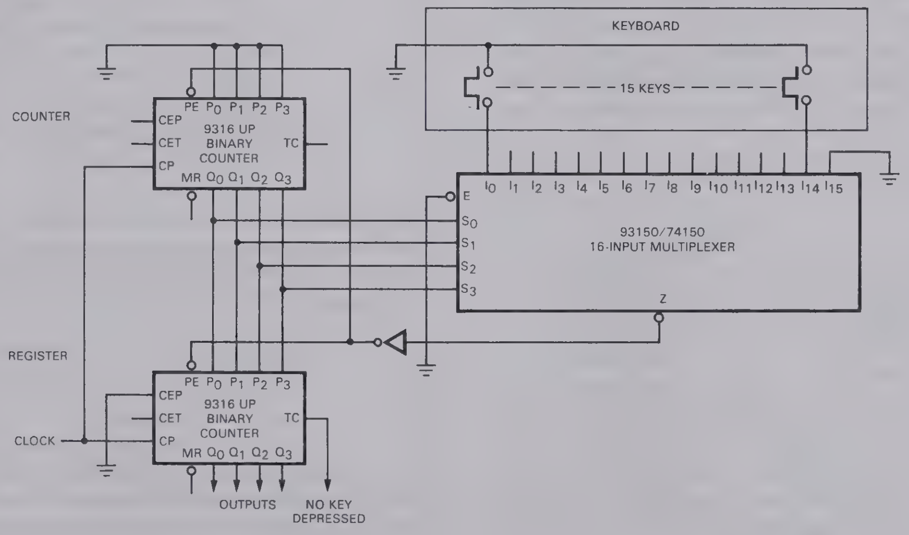
93150/74150 16-INPUT MULTIPLEXER APPLICATIONS

PATTERN GENERATOR


VCC
1 k
PE P0 P1 P2 P3
CEP
CET
9316 UP BINARY COUNTER
TC
CLOCK
CP
MR Q0 Q1 Q2 Q3
E
I0 I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13 I14 I15
S0
S1
S2
S3
93150 74150 16-INPUT MULTIPLEXER
Z
OUTPUT
ENABLE
JK
SYNCHRONIZED OUTPUT

Only two packages and sixteen single-pole, double-throw switches are used to implement at 16-bit pattern generator. The operation is self-explanatory. When the enable switch is Low, the output is the complement of the particular input of the 93150 multiplexer being addressed by the 9316 binary counter. A re-synchronizer may be added (1/2 dual J$\overline{K}$ or D flip-flop) to eliminate any decoding spikes.

15-KEY-TO-BINARY CODE CONVERTER


KEYBOARD
15 KEYS
COUNTER
PE P0 P1 P2 P3
CEP
CET
9316 UP BINARY COUNTER
TC
CP
MR Q0 Q1 Q2 Q3
E
I0 I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13 I14 I15
S0
S1
S2
S3
93150/74150 16-INPUT MULTIPLEXER
Z
REGISTER
PE P0 P1 P2 P3
CEP
CET
9316 UP BINARY COUNTER
TC
CLOCK
CP
MR Q0 Q1 Q2 Q3
OUTPUTS
NO KEY DEPRESSED

This scanning keyboard encoder generates a 4-bit binary coded output corresponding to the depressed key. The 9316 modulo 16 counter runs continuously, stepping the multiplexer and searching for a Low input. When a key is depressed, the multiplexer output goes Low (via an inverter), activating the Parallel Enable input of the second 9316 used as a 4-bit register. The corresponding counter code is loaded into this register, and the counter is reloaded with code zero. If several keys are depressed simultaneously, this encoder detects only the first one. Thus, it operates as a priority encoder and switch zero has the highest priority. When the keyboard is inactive address 15 selects the grounded input and the register inputs are all High. This state is internally decoded and activates the TC output.

MULTIPLEXERS AS FUNCTION GENERATORS

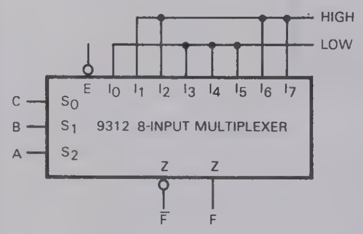
In most digital systems there are areas, usually in the control section, where a number of inputs generate an output in a highly irregular way. In other words, an unusual function must be generated which is apparently not available as an MSI building block.

In such cases, many designers tend to return to classical methods of logic design with NAND and NOR gates, using Boolean Algebra, Karnaugh maps and Veitch diagrams for logic minimization. Surprisingly enough, multiplexers can simplify these designs.

- The 9322 quad 2-input multiplexer can generate any *four* of the *16 different functions* of two variables.
- The 9309 and 93153 dual 4-input multiplexers can generate any *two* of the *256 different functions* of three variables.
- The 9312, 93151, and 93152 8-input multiplexers can generate any *one* of the *65,536 different functions* of four variables.
- The 93150 16-input multiplexer can generate any *one* of the over *4 billion different functions* of five variables.

If a function has a certain regularity, adders or a few NAND, NOR, AND, OR, exclusive-OR and inverter gates are possibly more economical. However, for a completely random function the multiplexer approach is more economical, certainly more compact and flexible, and easier to design.

Function generation with multiplexers is best explained with examples. An 8-input multiplexer such as the 9312 can obviously generate any possible function of three variables. The desired function is written as a truth table. The variables A, B, and C are applied to the Select inputs S_0, S_1, and S_2 and the eight inputs are connected to either a High or a Low level, according to the truth table. This method is simple, but *inefficient*.

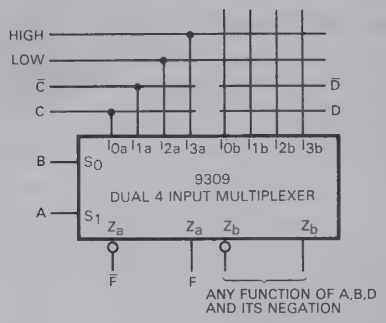

HIGH
LOW
C
B
A
E
S_0
S_1
S_2
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
9312 8-INPUT MULTIPLEXER
Z
Z
$\overline{F}$
F

A	B	C	F
L	L	L	L
L	L	H	H
L	H	L	H
L	H	H	L
H	L	L	L
H	L	H	L
H	H	L	H
H	H	H	H

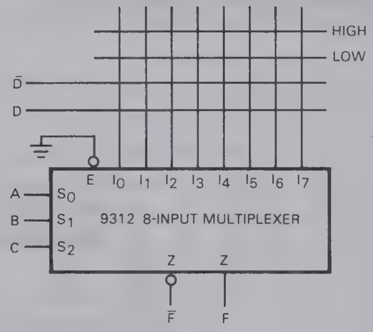
The same function can also be generated by one half of a dual 4-input multiplexer such as the 9309. For this purpose, the truth table is divided into four blocks as shown. Within each block, inputs A and B are constant, but output F can exhibit any of four characteristics.

- Low for both input codes independent of C.
- High for both input codes independent of C.
- Identical to C.
- Identical to $\overline{C}$.

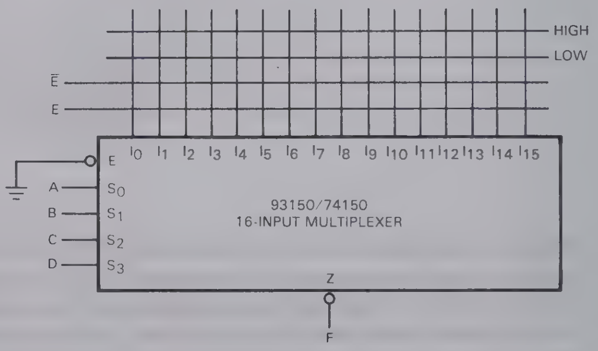
Therefore, the function can be implemented by a 4-input multiplexer, using the input variables A and B as Select inputs S_0 and S_1, and feeding the appropriate input with one of four signals: either a High, a Low, or the input variables C or $\overline{C}$. The other half of the 9309 can be used to generate any other function of the variables A, B, and any third variable, not necessarily C.


HIGH
LOW
$\overline{C}$
C
$\overline{D}$
D
B
A
S_0
S_1
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
9309
DUAL 4 INPUT MULTIPLEXER
Z_a
Z_a
Z_b
Z_b
$\overline{F}$
F
ANY FUNCTION OF A,B,D AND ITS NEGATION

The same reasoning can be applied to a function of four variables:


HIGH
LOW
$\overline{D}$
D
A
B
C
E
S_0
S_1
S_2
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
9312 8-INPUT MULTIPLEXER
Z
Z
$\overline{F}$
F

An 8-input multiplexer such as the 9312 can generate any of the 65,536 (2^{16}) possible functions of the four variables A, B, C, and D.


HIGH
LOW
$\overline{E}$
E
A
B
C
D
E
S_0
S_1
S_2
S_3
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 I_8 I_9 I_{10} I_{11} I_{12} I_{13} I_{14} I_{15}
93150/74150
16-INPUT MULTIPLEXER
Z
F

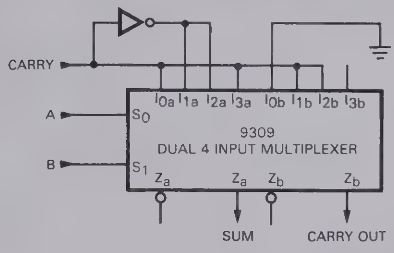
A 16-input multiplexer like the 93150 can generate any of the more than 4 billion (2^{32}) possible functions of five variables A, B, C, D, and E.

MULTIPLEXERS AS FUNCTION GENERATORS

FULL ADDER, FULL SUBTRACTOR

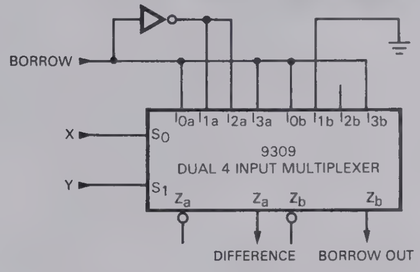
FULL ADDER (A plus B plus C_{IN})

INPUTS			OUTPUTS	
A	B	C_{IN}	S	C_O
L	L	L	L	L
L	L	H	H	L
H	L	L	H	L
H	L	H	L	H
L	H	L	H	L
L	H	H	L	H
H	H	L	L	H
H	H	H	H	H


CARRY
A
B
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309
DUAL 4 INPUT MULTIPLEXER
Za
Za
Zb
Zb
SUM
CARRY OUT

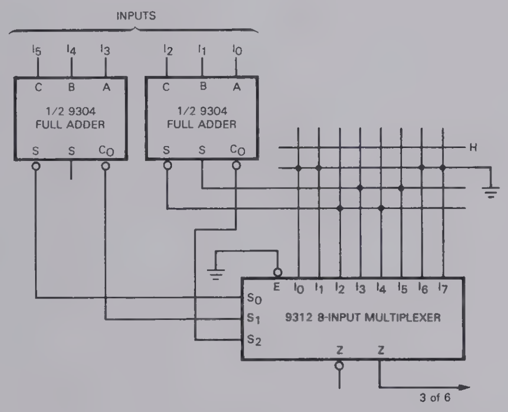
FULL SUBTRACTOR (X minus Y minus B_{IN})

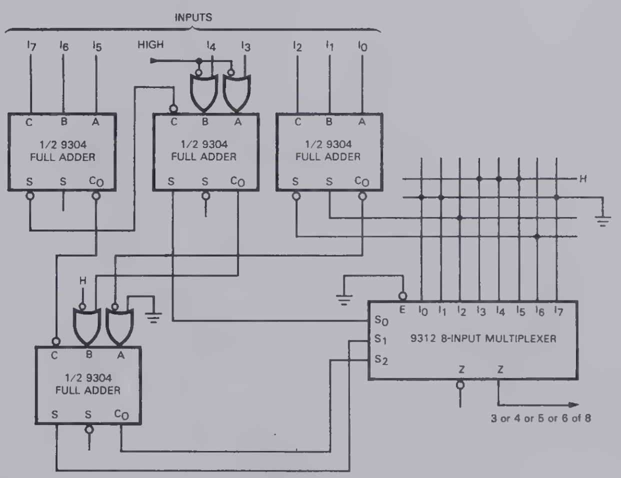
INPUTS			OUTPUTS	
X	Y	B_{IN}	D	B_O
L	L	L	L	L
L	L	H	H	H
H	L	L	H	L
H	L	H	L	L
L	H	L	H	H
L	H	H	L	H
H	H	L	L	L
H	H	H	H	H


BORROW
X
Y
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309
DUAL 4 INPUT MULTIPLEXER
Za
Za
Zb
Zb
DIFFERENCE
BORROW OUT

A 9309 dual 4-input multiplexer can implement any two functions of three variables. Therefore, it can be used as a full adder or as a full subtractor. These circuits demonstrate the versatility of the multiplexer as a function generator. However, the 9304 dual full adder and the 9383 4-bit adder are more efficient circuits for adding several bits in parallel.

X OF Y PATTERN DETECTOR


INPUTS
I5 I4 I3
I2 I1 I0
C B A
1/2 9304 FULL ADDER
S S Co
C B A
1/2 9304 FULL ADDER
S S Co
H
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
3 of 6


INPUTS
I7 I6 I5
HIGH
I4 I3
I2 I1 I0
C B A
1/2 9304 FULL ADDER
S S Co
C B A
1/2 9304 FULL ADDER
S S Co
C B A
1/2 9304 FULL ADDER
S S Co
H
H
C B A
1/2 9304 FULL ADDER
S S Co
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
3 or 4 or 5 or 6 of 8

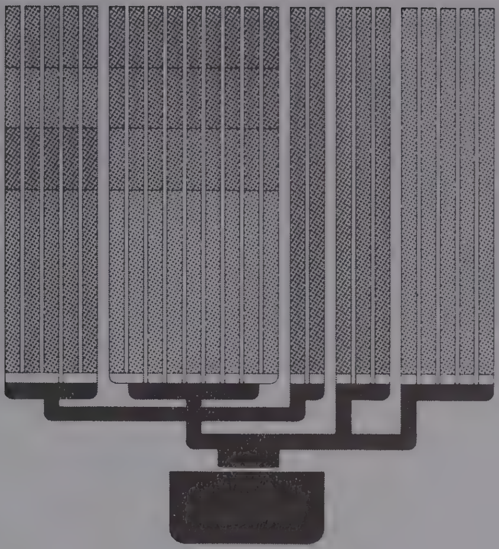
The detection of a specific number (or a specific set) of ones among many inputs is a common design problem, particularly with error correcting codes and when reading parallel data from multitrack digital tape decks and discs. A straightforward gate-minimized design is quite complex and usually inefficient. Multiplexers or adders can simplify such designs to some degree, but the most cost-efficient design uses a combination of both.

These designs use Full Adders to reduce the number of inputs to four variables, and then use an 8-input multiplexer to generate any desired function of these four variables. The result is an output which is High for a specified number (or specified set) of High inputs.

In the first example, two MSI packages (9304 + 9312) generate a High output when three (and only three) of the six inputs are High.

In the second example, three MSI circuits generate a High output when 3, 4, 5 or 6 of the eight inputs are High.

This combination of adders and multiplexers reduces package count to less than half of the equivalent conventional implementation. It also makes this circuit easily programmable for the detection of different patterns.



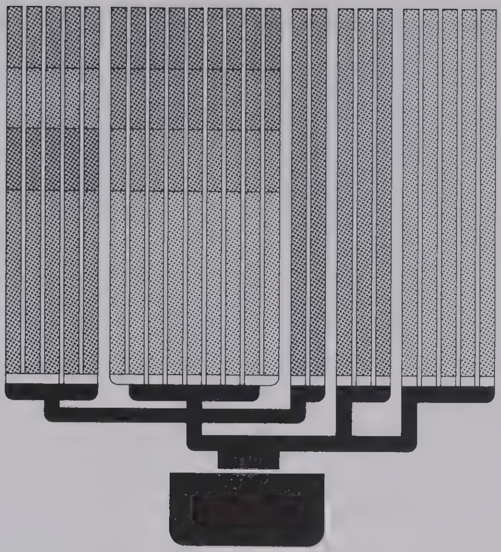
Decoders

1-of-64 Decoder
Demultiplexing
4-Phase Clock Generator
Function Generation
Address Decoding
Minterm Generation
Scanning Thumbwheel Switch Encoder
1-of-32 Decoder
Read Only Memory Control
Read/Write Memory Control
Decoding and Encoding
Switch Encoding
Data Demultiplexing
Clock Demultiplexing
Single-Master/Multiple-Slave Flip-Flop
Counters

DECODER SELECTION GUIDE

*One Enable Active High
**80 mA at V_{OL}
***Outputs Active High

Function	Device	Address Inputs	Active Low Enable	Active Low Outputs	Open Collector	Select Delay,ns (typ)	Enable Delay,ns (typ)	Power Dissipation mW(typ)	Fan Out UL	I_{MAX} mA	V_{MAX} V
Dual 1-of-4	9321	2 + 2	1 + 1	4 + 4		14.0	12	150	10.0		
	93L21	2 + 2	1 + 1	4 + 4		43.0	34	45	2.5		
	93S139	2 + 2	1 + 1	4 + 4		7.5	6	300	12.5		
	93155,74155	2	2 + 2	4 + 4		21.0	18	125	10.0		
	93156,74156	2	2 + 2	4 + 4	X	23.0	20	125		16.0	5.5
1-of-8	9301	3	1	8		22.0	22	145	10.0		
	93L01	3	1	8		36.0	36	45	2.5		
	9302	3	1	8	X	30.0	30	145		16.0	5.5
	9345,7445	3	1	8	X	40.0	40	215		20.0**	30.0
	93145,74145	3	1	8	X	40.0	40	215		20.0**	15.0
	9352,7442	3	1	8		26.0	26	140	10.0		
	93S138,74S138	3	3*	8		8.0	7	225	12.5		
	9334	3	1	8***		30.0	19	280	6.0		
1-of-10	9301	4 (BCD)		10		22.0		145	10.0		
	93L01	4 (BCD)		10		36.0		45	2.5		
	9302	4 (BCD)		10	X	30.0		145		16.0	5.5
	9345/7445	4 (BCD)		10	X	40.0		215		20.0	30.0
	93145,74145	4 (BCD)		10	X	40.0		215		20.0	15.0
	9352,7442	4 (BCD)		10		26.0		140	10.0		
	9353,7443	4 (Excess-3)		10		26.0		140	10.0		
	9354,7444	4 (Excess-3/Gray)		10		26.0		140	10.0		
1-of-16	9311,74154	4	2	16		21.0	17	175	10.0		
	93L11	4	2	16		70.0	48	58	2.5		



DECODERS

INTRODUCTION

There are two categories of decoders, logic decoders and display decoder/drivers. Logic decoders are MSI devices controlled by an address. They select and activate a particular output as specified by the address. Display decoders and display decoder/drivers generate numeric codes such as 7-segment and then provide the codes to a driver or drive the displays directly.

Logic decoders are the type discussed here and are available in many configurations as indicated by the selection guide. Logic decoders are used extensively in the selective addressing structures of memory systems. They are also used for data or clock routing, demultiplexing and can also act as minterm generators in random and control logic.

9321 DUAL I-OF-4 DECODER

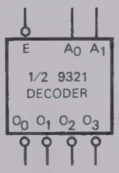
DESCRIPTION AND OPERATION

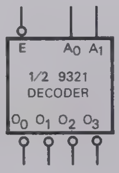
LEADS		LOADING
Decoder 1 and 2		
$\overline{E}$	Enable (Active Low) Input	1 UL
A_0, A_1	Address Inputs	1 UL
$\overline{O}_1$, $\overline{O}_1$, $\overline{O}_2$, $\overline{O}_3$	(Active Low) Outputs	10 UL

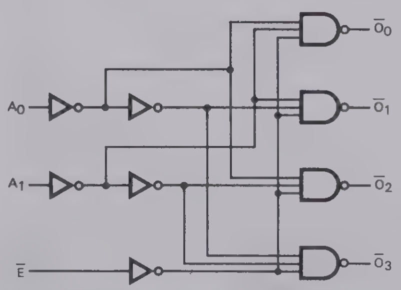
CHARACTERISTICS	
TYPICAL DELAY	A to Output 22 ns $\overline{E}$ to Output 15 ns
PACKAGE	16-Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	150 mW

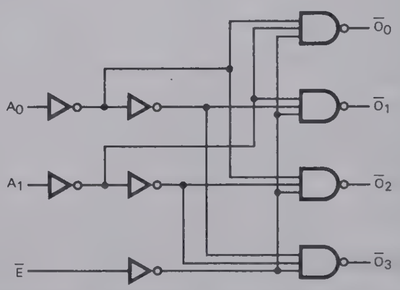
TRUTH TABLE

$\overline{E}$	A_0	A_1	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L





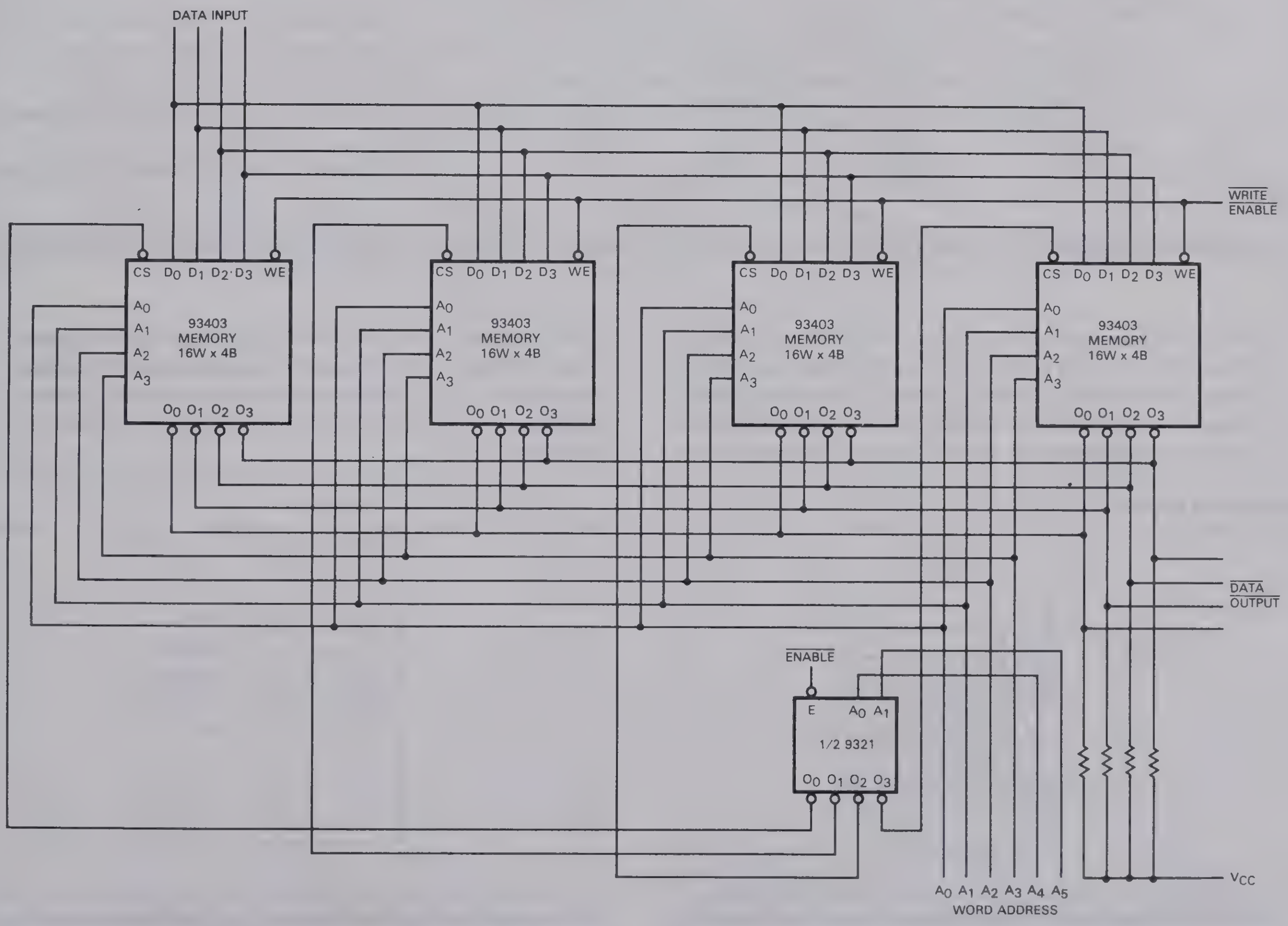




The 9321 is two separate decoders, each designed to accept two binary weighted inputs and provide four mutually exclusive active Low outputs as shown in the logic symbol. Each decoder can be used as a 4-output demultiplexer by using the Enable as a data input. Depending on the number of address bits that are changing simultaneously, the delay from address to output is equivalent to either two or three gate delays. The delay from Enable to output is always two gate delays, typically 15 ns.

9321 DUAL 1-OF-4 DECODER APPLICATIONS

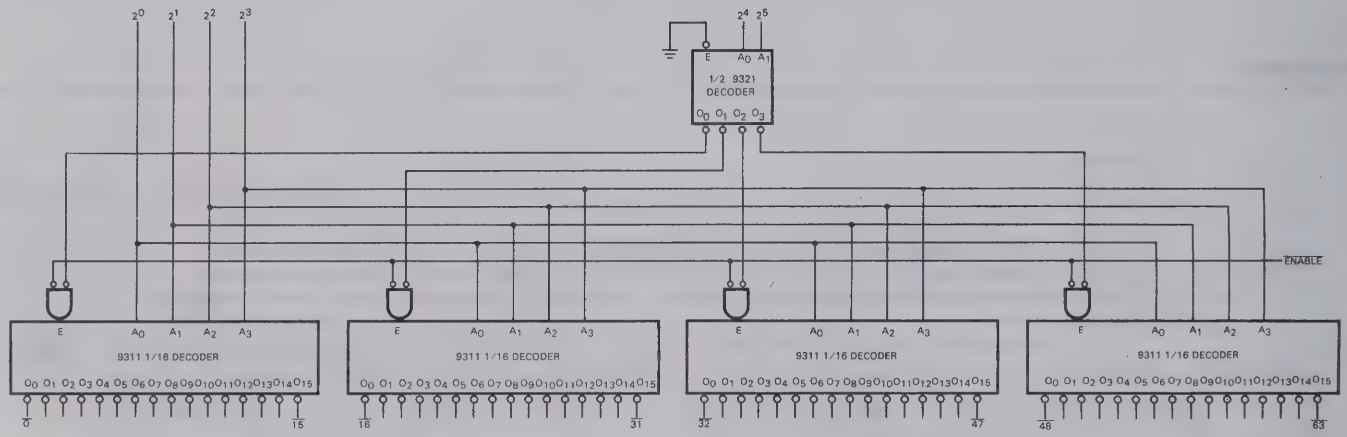
DECODING


DATA INPUT
WRITE ENABLE
CS D0 D1 D2 D3 WE
A0
A1
A2
A3
93403 MEMORY 16W x 4B
O0 O1 O2 O3
CS D0 D1 D2 D3 WE
A0
A1
A2
A3
93403 MEMORY 16W x 4B
O0 O1 O2 O3
CS D0 D1 D2 D3 WE
A0
A1
A2
A3
93403 MEMORY 16W x 4B
O0 O1 O2 O3
CS D0 D1 D2 D3 WE
A0
A1
A2
A3
93403 MEMORY 16W x 4B
O0 O1 O2 O3
DATA OUTPUT
ENABLE
E A0 A1
1/2 9321
O0 O1 O2 O3
VCC
A0 A1 A2 A3 A4 A5
WORD ADDRESS

The most obvious use of the 9321 is in logic decoding and memory addressing. As shown, the decoder supplies the extra decoding necessary to address a word in a 64-word semiconductor memory. One 1-of-4 decoder is used to decode the two most significant bits of memory address and to enable the appropriate memory units. The four least significant bits are decoded on the 93403. The high fan out capability of the 9321 allows it to drive ten 93403 memory units with a word length of 40 bits without additional buffers.

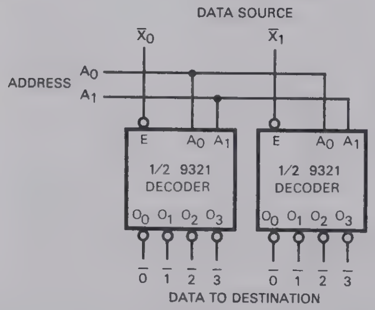
9321 DUAL 1-OF-4 DECODER APPLICATIONS

1-of-64 DECODER


2^0
2^1
2^2
2^3
2^4
2^5
E
A_0
A_1
1/2 9321
DECODER
O_0 O_1 O_2 O_3
ENABLE
E
A_0 A_1 A_2 A_3
9311 1/16 DECODER
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7 O_8 O_9 O_{10} O_{11} O_{12} O_{13} O_{14} O_{15}
$\overline{0}$
$\overline{15}$
$\overline{16}$
$\overline{31}$
$\overline{32}$
$\overline{47}$
$\overline{48}$
$\overline{63}$

The 9321 can be used to make a 1-of-64 decoder out of four 9311 1-of-16 decoders. Each of the four 9311s shown is selected by one of the 9321 decoder outputs. Thus, the two most significant bits are decoded by the 1-of-4 decoder and are used to select the appropriate 9311 decoder. The dual AND Enable of the 9311 permits the use of one Enable for selection and the other for strobing. It is preferable to frame decoder address changes at the last level for higher Enable switching speeds.

DEMULTIPLEXING

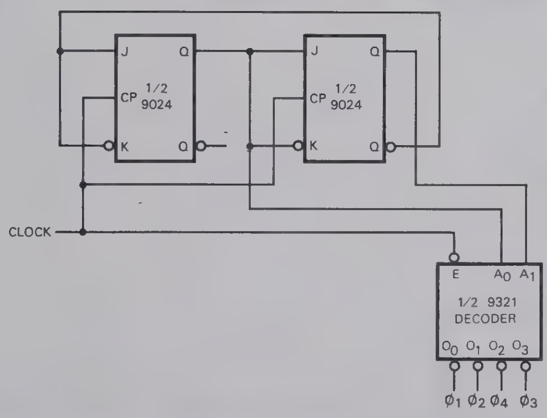

DATA SOURCE
$\overline{X}_0$
$\overline{X}_1$
ADDRESS
A_0
A_1
E
A_0 A_1
1/2 9321
DECODER
O_0 O_1 O_2 O_3
$\overline{0}$ $\overline{1}$ $\overline{2}$ $\overline{3}$
DATA TO DESTINATION

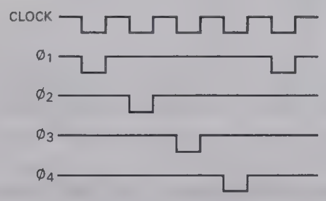
A_0	A_1	OUTPUT SELECTED
0	0	0
0	1	1
1	0	2
1	1	3

The 9321 can be used as a demultiplexer, routing data from a single source to a destination chosen by the applied address. The data is applied through the Enable and routed without inversion to the output specified by the address inputs A_0 and A_1. All unselected outputs remain High. For example, with both address inputs High, output 3 follows the state of the Enable input — Low when the Enable is Low and High when the Enable is High. Demultiplexing can be employed for either data routing or clock distribution. A 2-bit data demultiplexer is shown. Two bits of active Low data are routed to the outputs selected by the applied address as shown in the table above.

9321 DUAL 1-OF-4 DECODER APPLICATIONS

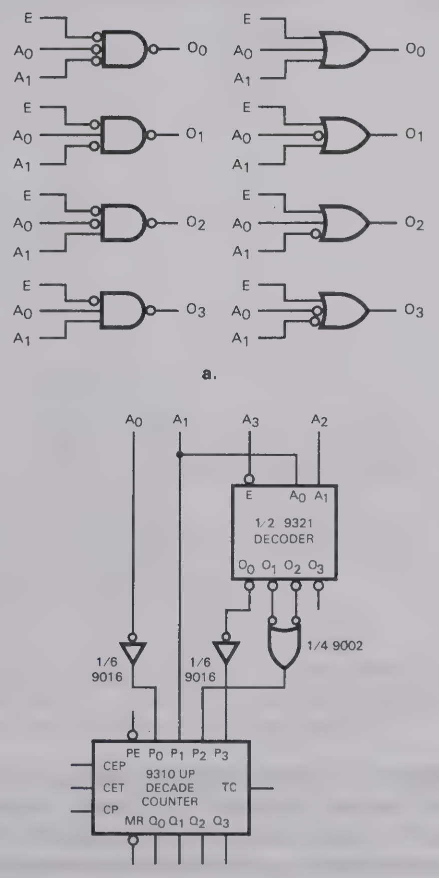
4-PHASE CLOCK GENERATOR


J
Q
CP
1/2
9024
K
Q
J
Q
CP
1/2
9024
K
Q
CLOCK
E
A0
A1
1/2 9321
DECODER
O0 O1 O2 O3
Ø1 Ø2 Ø4 Ø3



Clock demultiplexing for clock distribution and generation is readily accomplished with the 9321. This is a 4-phase clock generator producing non-overlapping clock pulses for TTL circuitry or to drive MOS circuitry through interfaces. Note that the Enable is used as the clock input, eliminating glitches by framing address changes which occur when the flip-flops, registers, or counters change state on the rising clock edge.

FUNCTION GENERATION


E
A0
A1
O0
O1
O2
O3
A0
A1
A3
A2
E
A0 A1
1/2 9321
DECODER
O0 O1 O2 O3
1/4 9002
1/6
9016
1/6
9016
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP
DECADE
COUNTER
TC
MR Q0 Q1 Q2 Q3

a.

b.

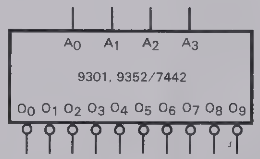
Each half of the 9321 generates all four minterms of two variables. These four minterms are useful in some applications, replacing logic functions and thereby reducing the number of packages required in a logic network. Gate functions which the 9321 can replace are shown in *a* above; *b* illustrates a nines complement circuit utilizing these gate functions.

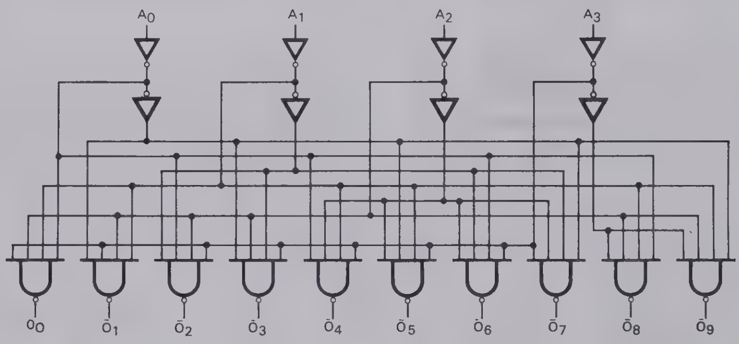
9301 AND 9352/7442 DECODER

DESCRIPTION AND OPERATION

LEADS		LOADING
A_0, A_1, A_2, A_3	Address Inputs	1 UL
$\bar{0}$ to $\bar{9}$	Outputs (Active Low)	10 UL

CHARACTERISTICS	
TYPICAL DELAY	A to Output 22 ns
PACKAGE	16 Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	145 mW





The 9301 and 9352/7442 1-of-10 decoders are high speed, complex function integrated circuits suitable for use in high speed digital equipment. These decoders have four inputs which act as an address to produce an output at the corresponding output terminal. They have high speed and excellent noise margins with reasonable power consumption and are compatible with other Fairchild MSI/TTL and DTL integrated circuits. With applications in a large number of areas, they considerably reduce integrated circuit package count, simplify packaging and increase system reliability.

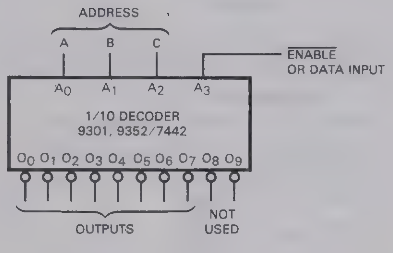
The 9301, 9352/7442 1-of-10 decoders accept four active BCD inputs and provides ten mutually exclusive active Low outputs. All outputs are High when binary codes greater than nine are applied to the inputs.

The most significant address input A_3 acts as active Low Enable or active High Inhibit input when the circuit is used as a 1-of-8 decoder or it can be used as the data input for a 8-output demultiplexer. Depending on the number of address bits that are changed simultaneously, the delay from address to output is equivalent to either two or three gate delays, typically 16 or 23 ns.

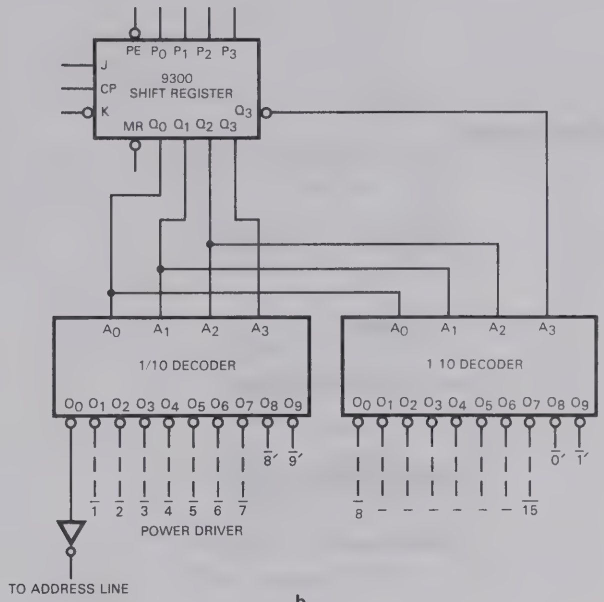
When the 9301, 9352/7442 are used as 1-of-8 decoders or demultiplexers, the three address inputs A_0 – A_2 can be interchanged and/or considered active Low, provided the outputs are relabeled appropriately. This may simplify printed circuit board layout.

9301 AND 9352/7442 DECODER APPLICATIONS

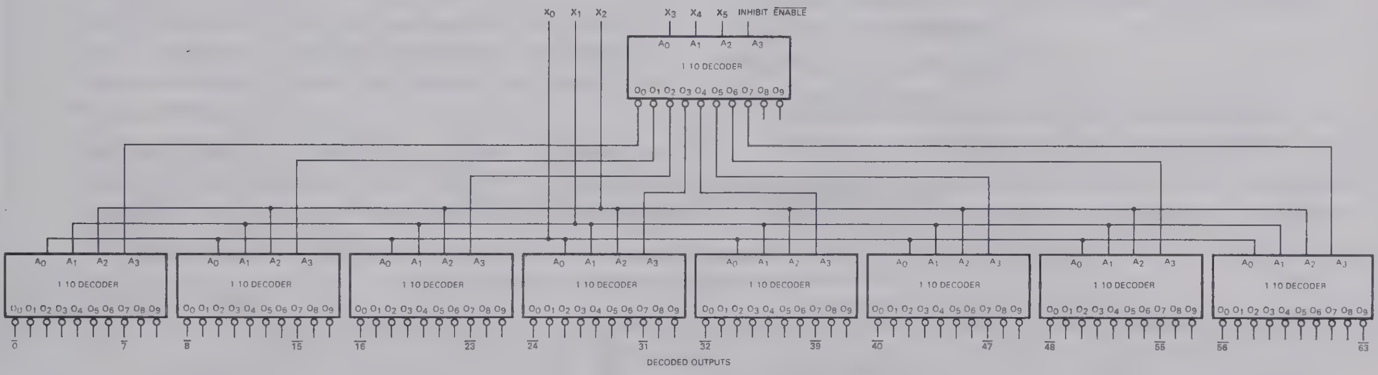
ADDRESS DECODING


ADDRESS
A
B
C
ENABLE OR DATA INPUT
A0 A1 A2 A3
1/10 DECODER
9301, 9352/7442
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
OUTPUTS
NOT USED

a.


PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
A0 A1 A2 A3
1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
A0 A1 A2 A3
1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
8′ 9′
0′ 1′
1 2 3 4 5 6 7
8 15
POWER DRIVER
TO ADDRESS LINE

b.


X0 X1 X2
X3 X4 X5 INHIBIT ENABLE
A0 A1 A2 A3
1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
0 7
8 15
16 23
24 31
32 39
40 47
48 55
56 63
DECODED OUTPUTS

c.

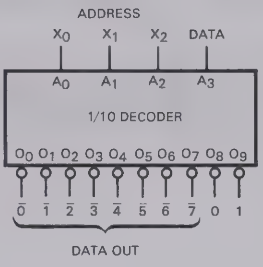
The 9301, 9352/7442 can be a 1-of-10 decoder, or a 1-of-8 decoder with Inhibit by using the A_3 input as a control line. Thus it can be considered a 1-of-8 decoder with active Low Enable or an 8-output demultiplexer using the A_3 input as data input. Note that all non-addressed outputs are High.

One method of decoding four binary digits to produce 16 outputs is shown in *b* above. A 9300 shift register holds the input variables with the most significant digit of the binary word on the right. The shift register has active High outputs from all stages as well as an active Low output from the last stage. The first three outputs of the register are inputs to the $A_0A_1A_2$ terminals of the decoders. The active High output of the last stage of the register is the A_3 input to the first decoder. The active Low output is the A_3 input to the second decoder. The outputs of the decoders drive inverting power gates which supply the fairly high currents required for addressing a memory of reasonable size. Alternate outputs (0′, 1′, 8′ and 9′) are available from the unused decimal outputs of the decoders.

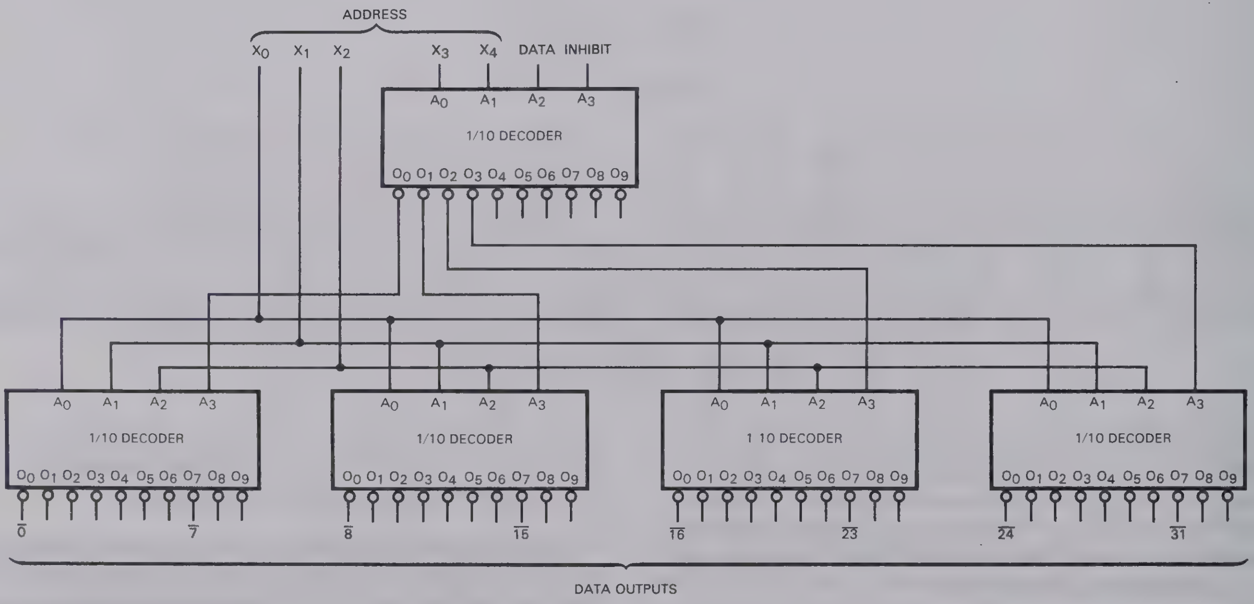
A 1-of-64 decoder (*c* above) can also be made by using several decoders. The three most significant bits of the input address are decoded in the first decoder to produce eight outputs which are then used as gating signals. The three least significant address bits are inputs to the $A_0A_1A_2$ terminals of a bank of eight decoders. Each decoder in the bank is selected by the appropriate output of the first, or gating, decoder via the A3 input terminal. This design can be extended to produce a 1-of-80 decoder by having four most significant address bits and by using the two outputs 8 and 9 from the gating decoder to control two additional decoders. This basic idea of using one or more decoders to select more decoders can be extended to decode even larger addresses.

9301 AND 9352/7442 DECODER APPLICATIONS

DEMULTIPLEXING


ADDRESS
X0 X1 X2 DATA
A0 A1 A2 A3
1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
0̄ 1̄ 2̄ 3̄ 4̄ 5̄ 6̄ 7̄ 0 1
DATA OUT

ADDRESS			OUTPUT SELECTED
X_0	X_1	X_2	
0	0	0	0
1	0	0	1
0	1	0	2
1	1	0	3
0	0	1	4
1	0	1	5
0	1	1	6
1	1	1	7

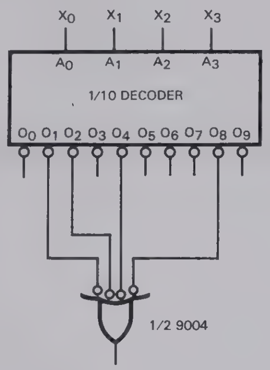

ADDRESS
X0 X1 X2 X3 X4
DATA INHIBIT
A0 A1 A2 A3
1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
0̄
7̄
8̄
1̄5̄
1̄6̄
2̄3̄
2̄4̄
3̄1̄
DATA OUTPUTS

The 9301 and 9352/7442 decoders can be used as 8-output demultiplexers. The first three inputs select the appropriate output; the logic level of the signal on the A_3 terminal determines its polarity. Therefore, data on the A_3 input is switched to the output terminal selected by the address, $A_0A_1A_2$. The last two outputs, 8 and 9, are the complements of the first two outputs, 0 and 1; note that data is not inverted when switched from the A_3 terminal to the selected output. The A_2 input of the decoder can also be used as a data input. In this mode, the A_3 terminal becomes an active High Inhibit and inputs A_0 and A_1 are a 2-bit address. Decoder outputs 0, 1, 2 and 3 are the assertion outputs of the demultiplexer and decoder outputs 4, 5, 6 and 7 are corresponding complements.

The multistage decoding scheme can also be used for demultiplexers requiring a large number of output channels. This design shows a 32-output demultiplexer. One decoder has two inputs to produce four active Low outputs, which are then used to select one of other four decoders. The remaining inputs of the first decoder are used as Data and Inhibit inputs.

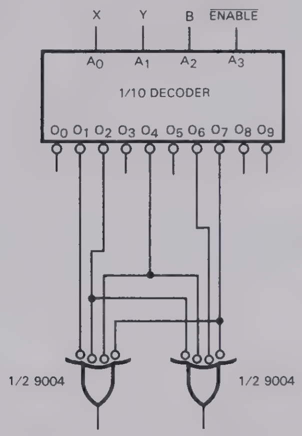
9301/9302 AND 9352/7442 DECODER APPLICATIONS

MINTERM GENERATION



SINGLE ONE DETECTOR

a.



DIFFERENCE BORROW

$\bar{E} \bullet$ (X minus Y minus B)

b.

The 9301 and 9352/7442 decoders can function as active Low output minterm generators producing the first 10 minterms of the 16 possible from four variables. The appropriate minterms can be summed with the use of an active Low input OR.

This technique is suitable for all types of control, sequencing, and decoding logic, and can considerably simplify the problem of generating a required sequence of outputs on a set of lines.

See the Multiplexer section for use of multiplexers as function generators.

The circuits above show

a) a single one detector (output F is High whenever one and only one of the X_0 – X_3 inputs is High) and

b) a gated full subtractor generating the Difference and the Borrow outputs of variables X, Y and B when the Enable input is Low.

INPUT-OUTPUT TABLE

				INPUT CODE															
A_0	A_1	A_2	A_3	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	2	4	8	0	1	2	3	4	5	6	7	8	9						
1	2	8	4	0	1	2	3	8	9			4	5	6	7				
1	4	8	2	0	1	8	9	2	3			4	5			6	7		
2	1	4	8	0	2	1	3	4	6	5	7	8		9					
2	1	8	4	0	2	1	3	8		9		4	6	5	7				
2	4	8	1	0	8	1	9	2		3		4		5		6		7	
4	1	2	8	0	2	4	6	1	3	5	7	8				9			
4	1	8	2	0	2	8		1	3	9		4	6			5	7		
4	2	8	1	0	8	2		1	9	3		4		6		5		7	
8	1	2	4	0	2	4	6	8				1	3	5	7	9			
8	1	4	2	0	2	8		4	6			1	3	9		5	7		
8	2	4	1	0	8	2		4		6		1	9	3		5		7	
INPUTS APPLIED				OUTPUTS ACTIVATED															

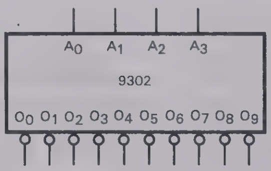
The 9301/9302 decodes 10 of the possible 16 minterms of the four variables applied to the inputs A_0 – A_3. The table above shows how different groups of minterms can be decoded by changing the assignment of the 1248 input signals to the A_0 – A_3 address inputs. Interconnection of outputs can generate many different functions of four variables.

9302 1-OF-10 DECODER

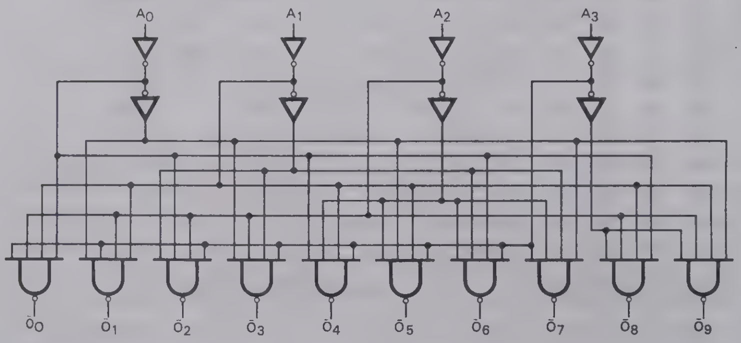
DESCRIPTION AND OPERATION

LEADS		LOADING
A_0, A_1, A_2, A_3	Address Inputs	1 UL
$\bar{0}$ to $\bar{9}$	Outputs (Active Low) Open Collector	10 UL

CHARACTERISTICS	
TYPICAL DELAY	A to Output 22 ns
PACKAGE	16 Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	145 mW



A0 A1 A2 A3
9302
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9

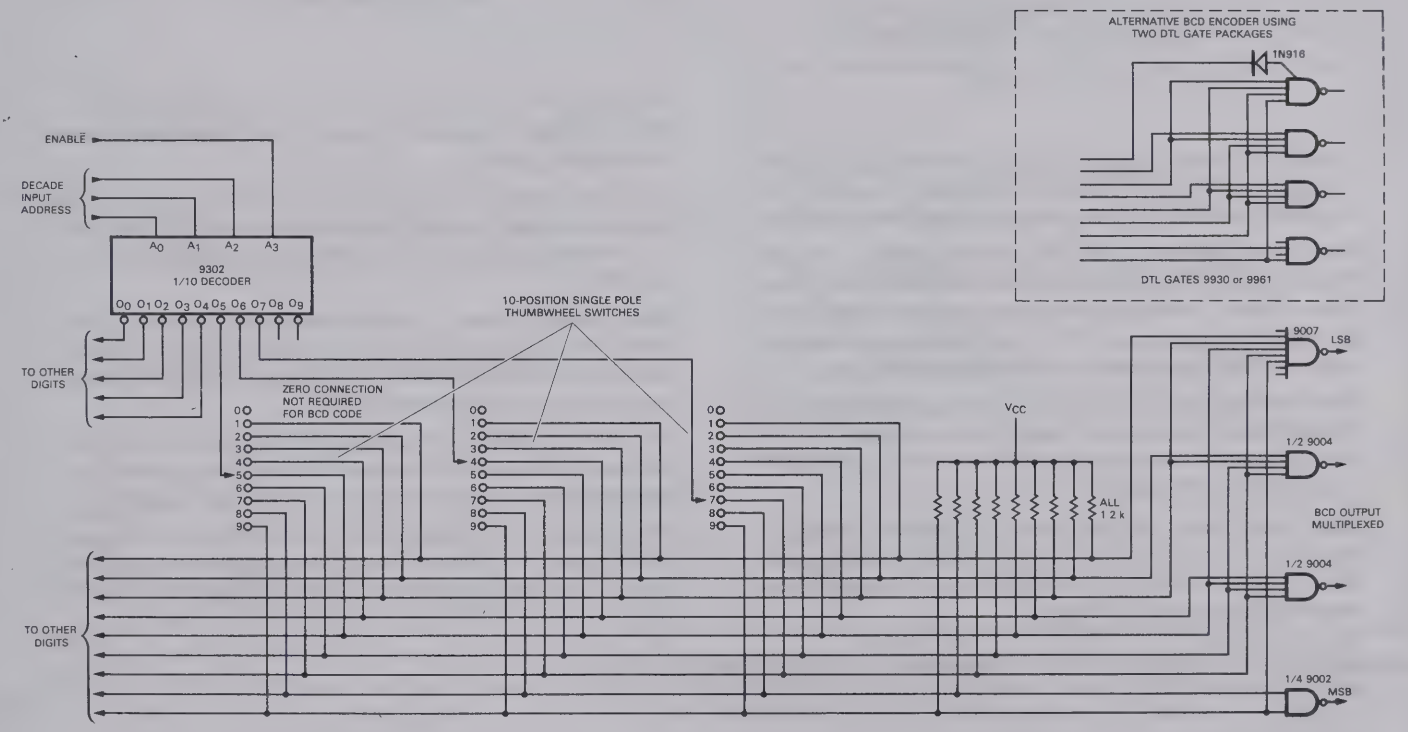


A0 A1 A2 A3
Ō0 Ō1 Ō2 Ō3 Ō4 Ō5 Ō6 Ō7 Ō8 Ō9

The 9302 is functionally and lead identical to the 9301 1-of-10 decoder, but has open collector outputs. These can therefore be wired together to simplify function generation and provide output bussing.

9302 1-OF-10 DECODER APPLICATIONS

A SCANNING THUMBWHEEL SWITCH ENCODER


ENABLE
DECADE INPUT ADDRESS
A0 A1 A2 A3
9302 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
TO OTHER DIGITS
ZERO CONNECTION NOT REQUIRED FOR BCD CODE
10-POSITION SINGLE POLE THUMBWHEEL SWITCHES
0 1 2 3 4 5 6 7 8 9
VCC
ALL 1.2 k
ALTERNATIVE BCD ENCODER USING TWO DTL GATE PACKAGES
1N916
DTL GATES 9930 or 9961
9007
LSB
1/2 9004
BCD OUTPUT MULTIPLEXED
1/2 9004
1/4 9002
MSB
TO OTHER DIGITS

Thumbwheel switches are becoming increasingly popular for remote programming of counters, displays, industrial control systems, etc. To reduce the number of interconnections between the switches and the destination, it is desirable to use multiplexing techniques. Ten decades of BCD thumbwheel switches unmultiplexed would require > 40 interconnections, while a multiplexed system requires < 20 interconnections.

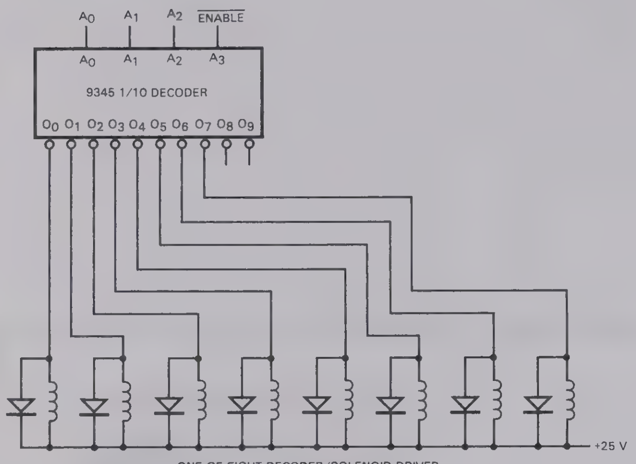
The conventional method of multiplexing uses BCD (or any 4-bit code) thumbwheel switches, each with a diode in series with the four outputs. These are connected to four parallel bus lines to the system output. The wiper arm of each switch is then selected from a decoder. Since the code is generated by the switch, this conventional system requires different thumbwheel switches for different codes, some of which are considerably more expensive than others e.g. nines complement.

The system described here requires no diodes and uses standard, low cost, single pole decade switches. The ten outputs are bussed to a simple encoder that generates the code required; the schematic shows BCD nines complement is equally simple.

The wiper arm of each switch is separately addressed by the active Low output of the 9302 open collector decoder. Nine pull up resistors at the encoder inputs insure proper noise immunity. Open collector decoder outputs are required since two or more switches might be in the same position, thus interconnecting several decoder outputs. The address applied to the decoder determines which switch is addressed; its position appears at the outputs of the four NAND gates. This system uses fewer and simpler parts and fewer solder joints than a conventional system, providing improved reliability.

9345/9352/9353/9354 1-OF-10 DECODERS

DESCRIPTIONS



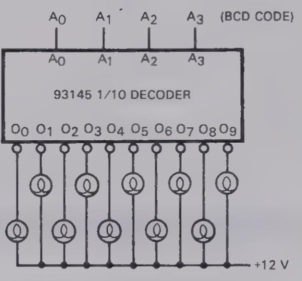
ONE-OF-EIGHT DECODER/SOLENOID DRIVER

The 9345/7445 and the 93145/74145 are 1-of-10 decoders with open collector outputs that can sink more current and withstand more voltage than the 9302.

These decoders accept BCD inputs on the A_0 – A_3 address lines and generate ten mutually exclusive active Low outputs. When an input code greater than nine is applied, all outputs are off. These devices can thus be used as 1-of-8 decoders with active Low Enable.

Both devices can sink 20 mA while maintaining the standardized guaranteed output Low voltage (V_{OL}) of 0.4 V, but they can sink up to 80 mA with a guaranteed V_{OL} of less than 0.9 V.

The 9345/7445 has an output breakdown voltage of 30 V, while the 93145/74145 has an output breakdown voltage of 15 V. These devices are ideally suited as lamp and solenoid drivers.



ONE-OF-TEN DECODER/LAMP DRIVER

The 9352/7442 is a 1-of-10 decoder, logically identical to the 9301 but with different lead assignment and somewhat slower.

The 9353/7443 is an Excess-3 to 1-of-10 decoder that accepts Excess-3 coded inputs (binary range 3 through 12) and generates ten mutually exclusive active Low outputs.

The Excess-3 code is used in decimal arithmetic because of its self-complementing feature (the bit-wise complement of a number is also the nines complement) which simplifies subtraction.

The 9354/7444 is an Excess-3 Gray code to 1-of-10 decoder that accepts Excess-3 Gray coded inputs and generates ten mutually exclusive active Low outputs.

The Excess-3 Gray code is used in decimal position encoders, since it retains the characteristics of a Gray code (only one bit changes between adjacent states) even on the change between 9 and 0.

DECODER OUTPUT STATES FOR COMMONLY USED INPUT CODES

DECIMAL VALUE		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
INPUTS	A_0	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	
	A_1	L	L	H	H	L	L	H	H	L	L	H	H	L	L	H	H	
	A_2	L	L	L	L	H	H	H	H	L	L	L	L	H	H	H	H	
	A_3	L	L	L	L	L	L	L	L	H	H	H	H	H	H	H	H	
BCD — 9301, 9352 TTL Outputs 9302 Open Collector Outputs 9345, 93145 High Current, High Voltage, Open Collector Outputs		0	1	2	3	4	5	6	7	8	9	–	–	–	–	–	–	ACTIVE OUTPUTS (LOW)
EXCESS-3 9353 TTL Outputs		–	–	–	0	1	2	3	4	5	6	7	8	9	–	–	–	
EXCESS-3 GRAY 9354 TTL Outputs		–	–	0	–	4	3	1	2	–	–	9	–	5	6	8	7	

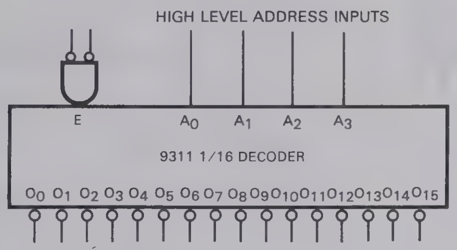
This is a consolidated truth table for all Fairchild TTL 1-of-10 decoders. A given set of inputs activates different output leads depending on decoder types used, BCD, EXCESS 3, or EXCESS-3 Gray.

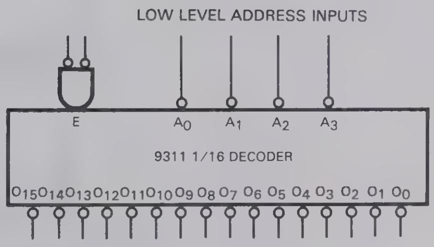
9311 1-OF-16 DECODER

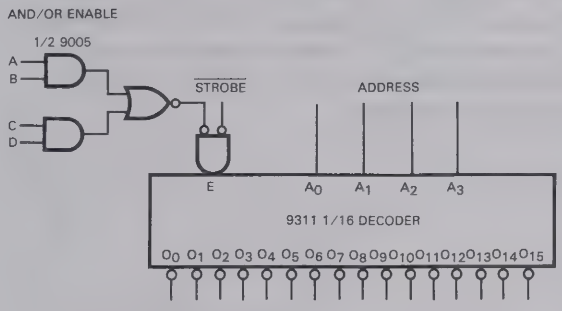
DESCRIPTION AND OPERATION

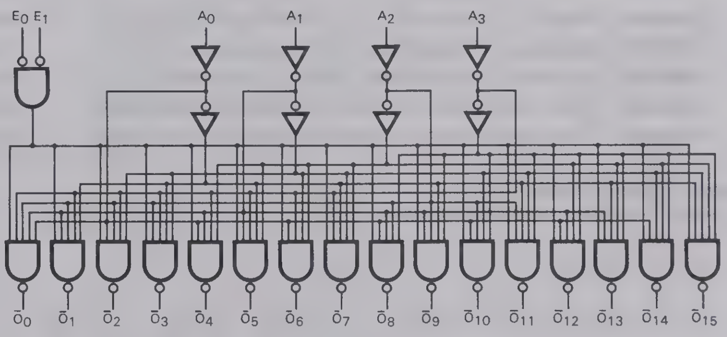
LEADS		LOADING
A_0, A_1, A_2, A_3	Address Inputs	1 UL
$\overline{E}_0, \overline{E}_1$	AND Enable (Active Low) Inputs	1 UL
$\overline{0}$ to $\overline{15}$	(Active Low) Outputs	10 UL

CHARACTERISTICS	
TYPICAL DELAY	A to Output 21 ns $\overline{E}$ to Output 17 ns
PACKAGE	24 Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	175 mW


HIGH LEVEL ADDRESS INPUTS
E
A0
A1
A2
A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15


LOW LEVEL ADDRESS INPUTS
E
A0
A1
A2
A3
9311 1/16 DECODER
O15 O14 O13 O12 O11 O10 O9 O8 O7 O6 O5 O4 O3 O2 O1 O0


AND/OR ENABLE
1/2 9005
A
B
C
D
STROBE
ADDRESS
E
A0
A1
A2
A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15


E0 E1
A0
A1
A2
A3
Ō0 Ō1 Ō2 Ō3 Ō4 Ō5 Ō6 Ō7 Ō8 Ō9 Ō10 Ō11 Ō12 Ō13 Ō14 Ō15

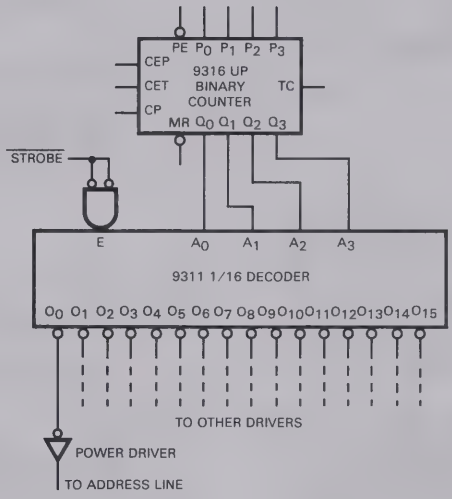
The 9311 is a 1-of-16 decoder with two active Low Enables in a 24-lead Dual In-Line package. All leads are used to provide maximum flexibility and functional capability. It has buffered inputs to reduce input loading and uses NAND gates to produce the 16 possible outputs from four address inputs. One particular output goes Low when its address is applied to the inputs $A_0 - A_3$ and both Enable inputs are Low. This 2-input active Low Enable gate increases logic flexibility. The 9311 can be controlled by other decoders or gates and still retain an overriding active Low Enable input, or a matrix of 9311 can be controlled by two decoders or NAND gates, one in the x, the other in the y direction.

Depending on the number of address bits that are changing simultaneously, the delay from address to output is equivalent to either two or three gate delays. The delay from Enable to output is always two gate delays, typically 14 ns. The four address inputs have 16 different states, each activating one particular output. The logic symbol is normally drawn for active High address inputs, but it can equally well be drawn for active Low address inputs, in which case the outputs have to be relabeled as shown.

The 2-input active Low Enable AND gate can also be considered a 2-input active High Inhibit OR-gate. Logic power of the active Low AND Enable gate is shown. The decoder is enabled only during an AND/OR condition. The remaining input on the AND Enable gate can then serve as an active Low strobe signal.

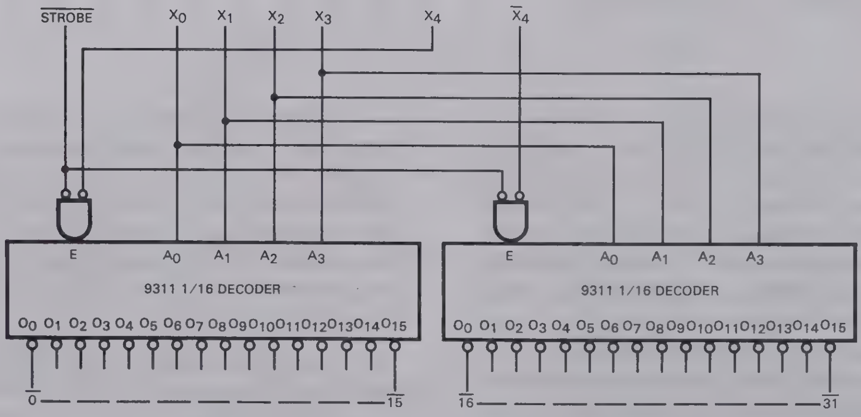
9311 1-OF-16 DECODER APPLICATIONS

ADDRESS DECODING


PE
P_0
P_1
P_2
P_3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR
Q_0
Q_1
Q_2
Q_3
STROBE
E
A_0
A_1
A_2
A_3
9311 1/16 DECODER
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7 O_8 O_9 O_{10} O_{11} O_{12} O_{13} O_{14} O_{15}
TO OTHER DRIVERS
POWER DRIVER
TO ADDRESS LINE

Address decoding is one of the prime applications of the 9311 decoder in core, thin film, and semiconductor memory systems. The decoder does not normally have enough drive capability to drive memory elements directly. Therefore, some form of buffer is often used between the decoder outputs and the memory cells. Since these buffers are generally inverting units, the active Low level outputs of the decoder are an advantage. The address decoder above uses a 9316 4-bit binary counter as a holding register.

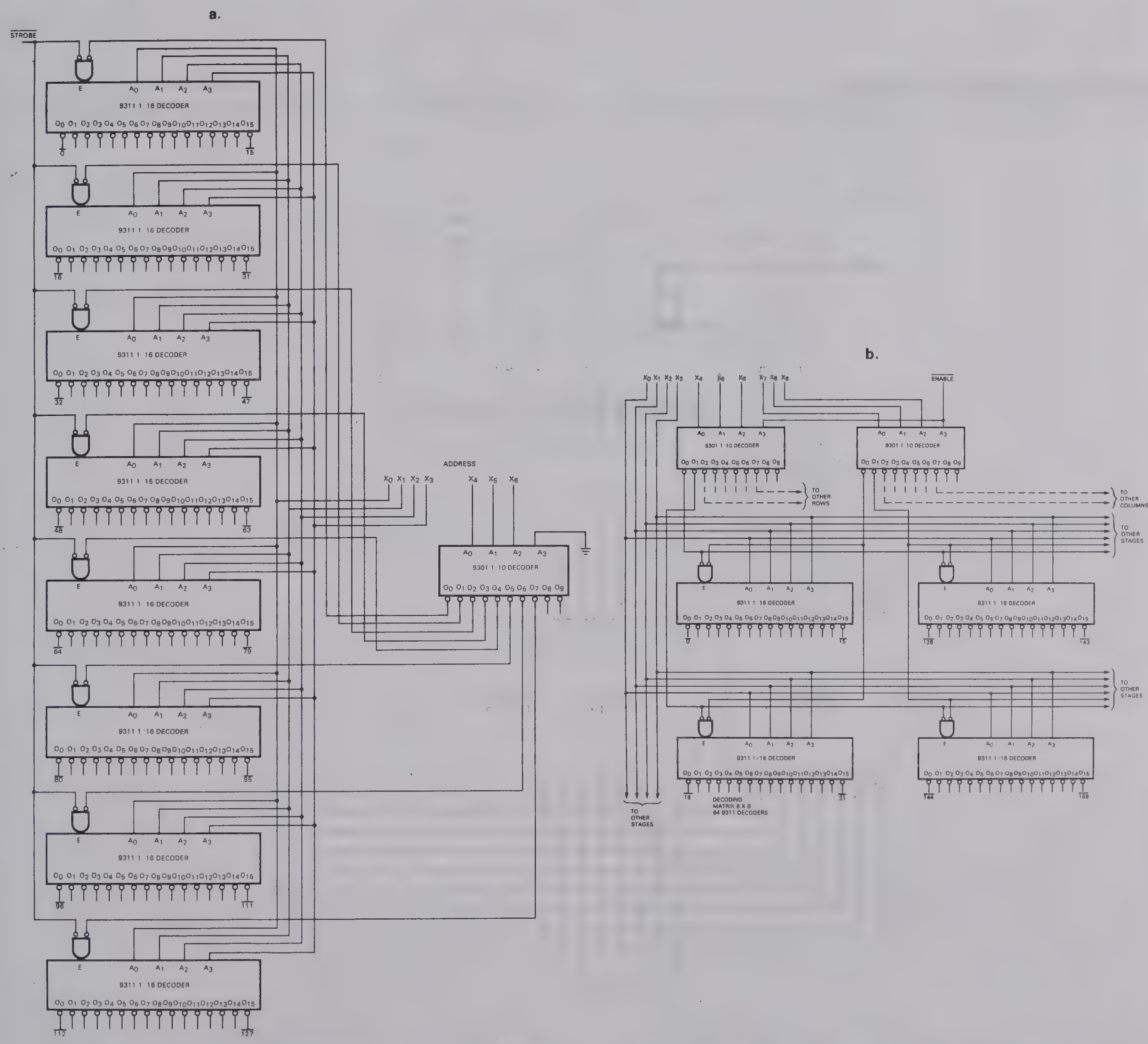
1-OF-32 DECODER


STROBE
X_0
X_1
X_2
X_3
X_4
$\overline{X}_4$
E
A_0
A_1
A_2
A_3
9311 1/16 DECODER
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7 O_8 O_9 O_{10} O_{11} O_{12} O_{13} O_{14} O_{15}
$\overline{0}$
$\overline{15}$
E
A_0
A_1
A_2
A_3
9311 1/16 DECODER
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7 O_8 O_9 O_{10} O_{11} O_{12} O_{13} O_{14} O_{15}
$\overline{16}$
$\overline{31}$

Several 9311 decoders can be operated together to decode words of more than four bits; the example above shows two 9311 decoders decoding five binary digits.

9311 1-OF-16 DECODER APPLICATIONS

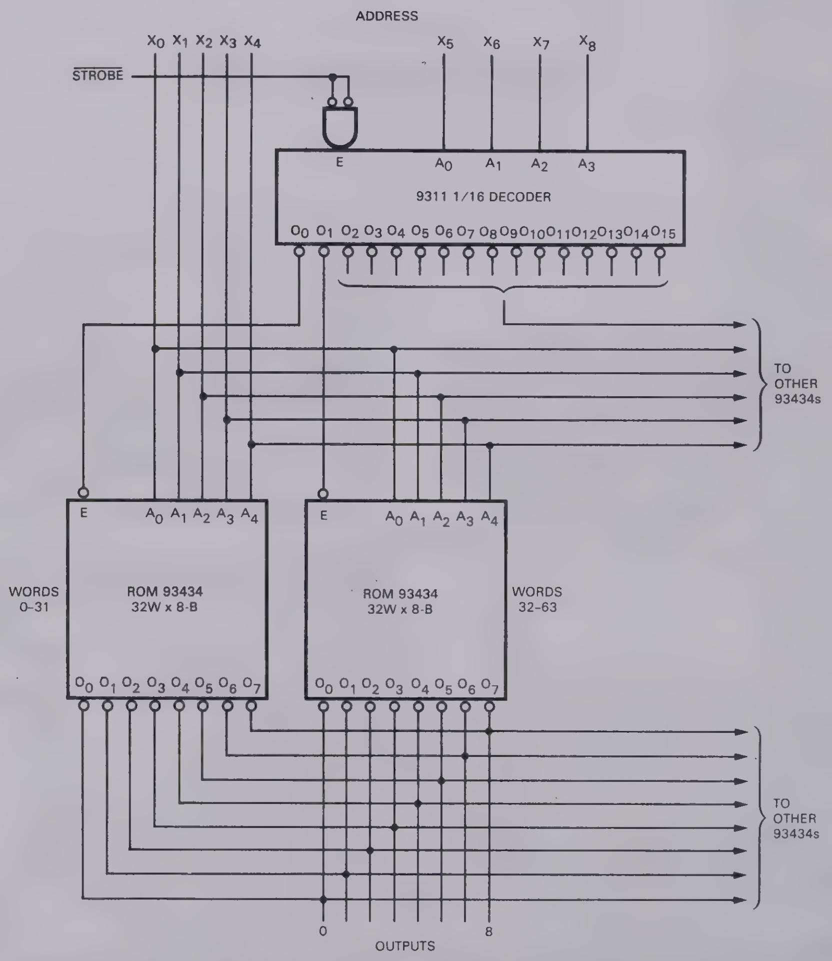
LARGE ADDRESS DECODING


a.
STROBE
E
A0 A1 A2 A3
9311 1 16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
0
15
16
31
32
47
48
63
64
79
80
95
96
111
112
127
ADDRESS
X0 X1 X2 X3
X4 X5 X6
9301 1 10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
b.
X0 X1 X2 X3 X4 X5 X6 X7 X8 X9
ENABLE
TO OTHER ROWS
TO OTHER COLUMNS
TO OTHER STAGES
128
143
144
159
TO OTHER STAGES
DECODING MATRIX 8 X 8
64 9311 DECODERS

The decoding scheme in *a* uses a 9301 1-of-10 decoder to select a particular 9311. The address inputs of the 9301 represent the three most significant address bits and the address inputs of the 9311 decoders commonly represent the four least significant address bits. A 7-bit binary address is decoded to 128 output lines, and the second AND Enable input is used as a strobe. An even larger 1-of-1024 decoding scheme, using an XY matrix approach, is shown in *b*.

9311 1-OF-16 DECODER APPLICATIONS

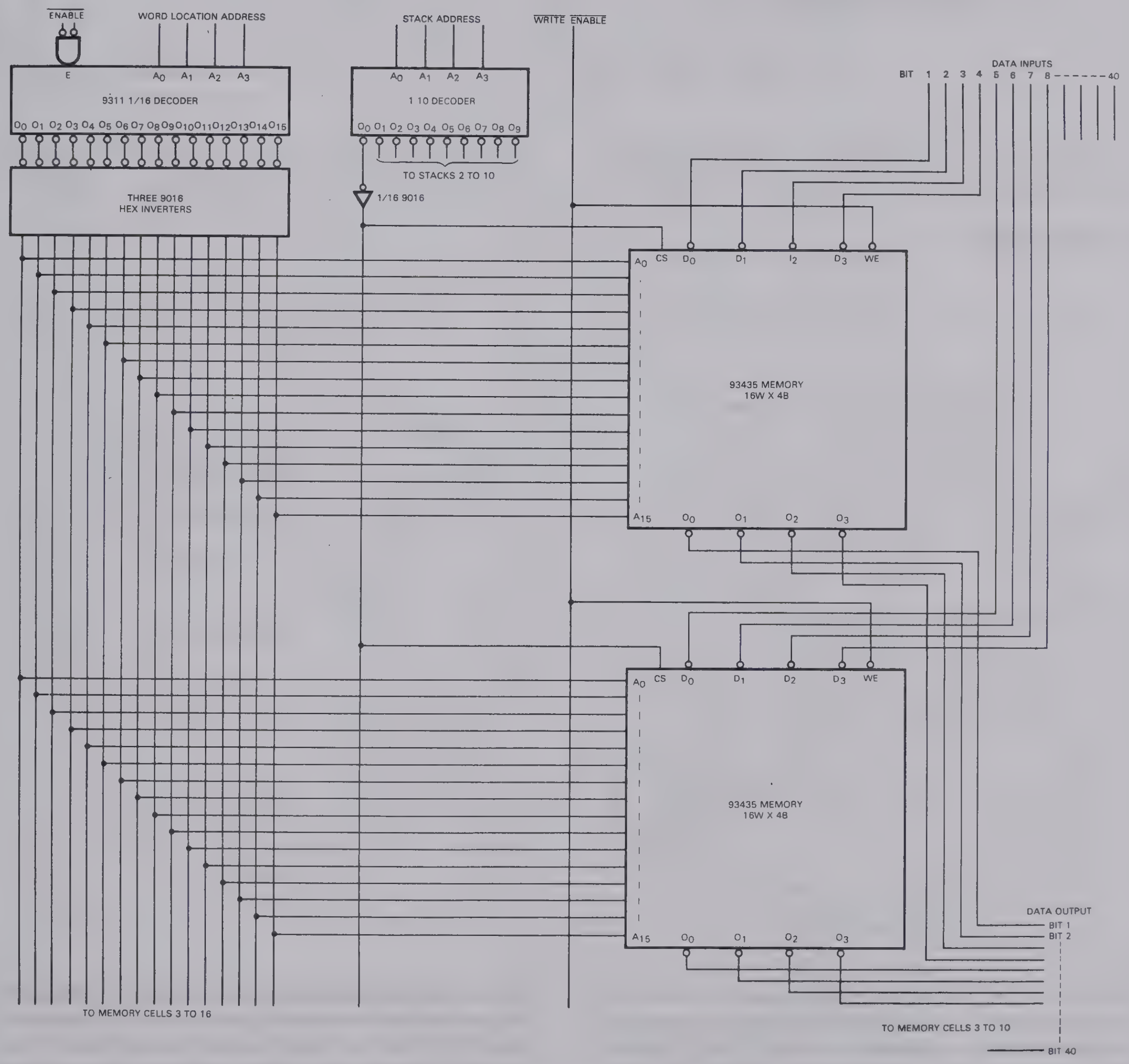
READ ONLY MEMORY CONTROL


ADDRESS
X0 X1 X2 X3 X4
X5 X6 X7 X8
STROBE
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
TO OTHER 93434s
E
A0 A1 A2 A3 A4
WORDS 0–31
ROM 93434 32W x 8-B
O0 O1 O2 O3 O4 O5 O6 O7
E
A0 A1 A2 A3 A4
ROM 93434 32W x 8-B
WORDS 32–63
O0 O1 O2 O3 O4 O5 O6 O7
TO OTHER 93434s
0
8
OUTPUTS

A 9311 decoder can select a particular bipolar read only memory from a group of memories (93434s) comprising a complete stack. A single 9311 can control a group of 16 of these memories. Each memory contains 256 bits arranged in a 32-word x 8-bit format, with the outputs of common digits OR tied. An output from the 9311 can drive up to ten 93434 memories, allowing control of a 512-word x 80-bit memory or 40,960 bits of information with a single decoder.

9311 1-OF-16 DECODER APPLICATIONS

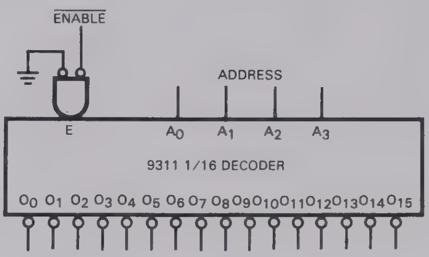
READ/WRITE MEMORY CONTROL


ENABLE
WORD LOCATION ADDRESS
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
THREE 9016
HEX INVERTERS
STACK ADDRESS
A0 A1 A2 A3
1 10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
TO STACKS 2 TO 10
1/16 9016
WRITE ENABLE
DATA INPUTS
BIT 1 2 3 4 5 6 7 8 ------ 40
A0 CS D0 D1 I2 D3 WE
93435 MEMORY
16W X 4B
A15 O0 O1 O2 O3
A0 CS D0 D1 D2 D3 WE
93435 MEMORY
16W X 4B
A15 O0 O1 O2 O3
DATA OUTPUT
BIT 1
BIT 2
TO MEMORY CELLS 3 TO 10
BIT 40
TO MEMORY CELLS 3 TO 16

Another memory configuration incorporating the 9311 decoder is built around the 93435 16-word x 4-bit read/write memory. The design shows a typical semiconductor memory stack of 160-words x 40-bits. The 93435 is a linear select device requiring both active high inputs and inverting buffers between the decoders and memory cells as shown above. Since the decoder outputs can drive ten inverters, and each of these can drive ten 93435 memories, a single 9311 with a 9301 can control a fairly large semiconductor random access read/write memory.

9311 1-OF-16 DECODER APPLICATIONS

DECODING AND ENCODING

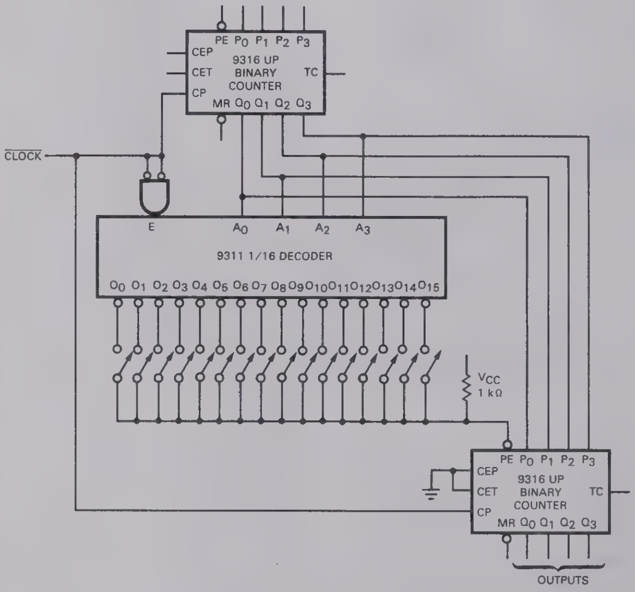


ENABLE
ADDRESS
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15

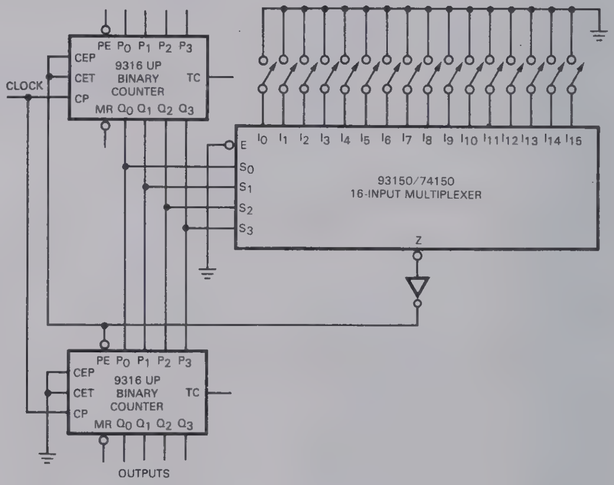
DECIMAL DIGIT	OUTPUT ACTIVATED			
	8421	5421	EXCESS 3	GRAY
0	0	0	3	0
1	1	1	4	1
2	2	2	5	3
3	3	3	6	2
4	4	4	7	6
5	5	8	8	7
6	6	9	9	5
7	7	10	10	4
8	8	11	11	12
9	9	12	12	13

The 9311 decoder can decode any 4-bit weighted or unweighted code simply by selecting the appropriate outputs in the desired sequence. Decoding for several well-known 4-bit codes is shown above.

SWITCH ENCODING



PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
VCC
1 kΩ
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
OUTPUTS



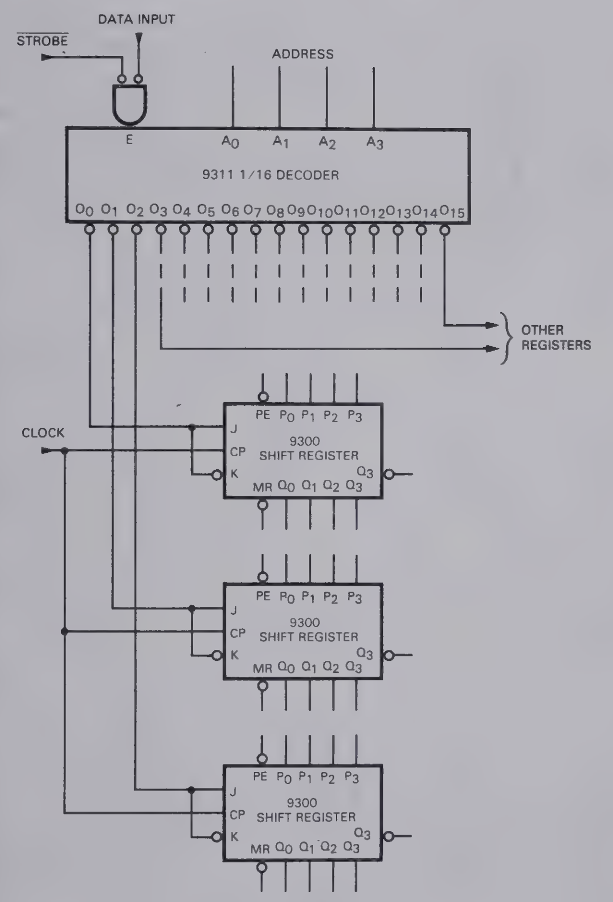
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
E
I0 I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13 I14 I15
S0
S1
S2
S3
93150/74150 16-INPUT MULTIPLEXER
Z
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
OUTPUTS

The 9311 1-of-16 decoder can be used in a scanning switch encoder. A 9316 modulo 16 free running synchronous counter is decoded by the 9311, and 16 push-button switches are connected to its outputs. The clock is connected to the Enable inputs to prevent output glitches when the counter outputs change after the rising clock edge. This scheme assumes that only one switch can be activated at a time. The common point of all switches is normally High, going Low when the position of the activated switch is decoded, which loads the counter contents into the synchronous holding register, another 9316 counter used in parallel load mode.

When it is possible for several switches to be activated simultaneously, it is better to take a different approach using a 16-input multiplexer controlled by a free running counter. The output of the multiplexer controls the loading of the counter contents into the synchronous holding register, consisting of another 9316 used only in the parallel load mode. The counter stops whenever a closed switch is decoded, thus ignoring further switch closures until the first switch has been released. This feature is called 2-key roll over.

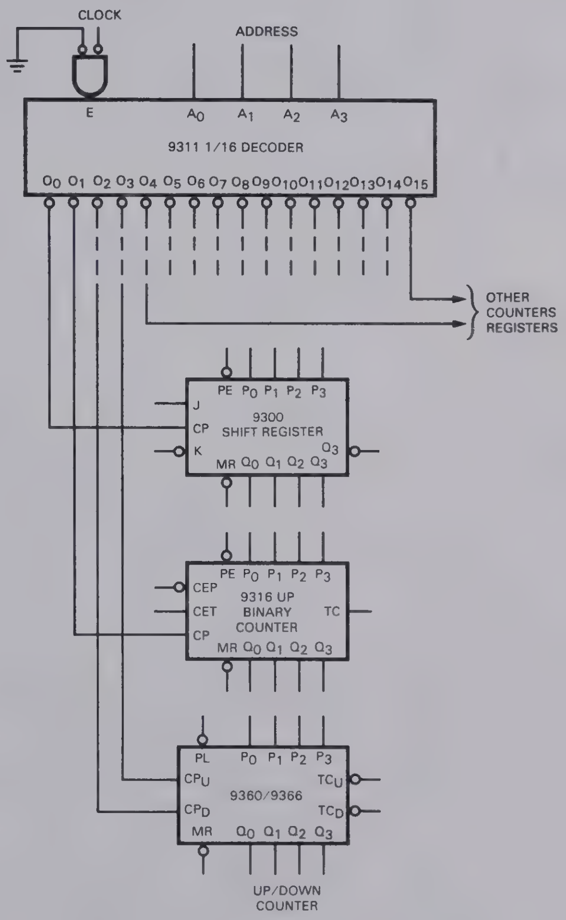
9311 1-OF-16 DECODER APPLICATIONS

DATA DEMULTIPLEXING


DATA INPUT
STROBE
ADDRESS
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
OTHER REGISTERS
CLOCK
PE P0 P1 P2 P3
J
CP
K
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3

The 9311 decoder can select a particular output under the control of an address; and the active Low Enable can be used as a data input that is routed to a specified output under control of the address input. If the address configuration selects output zero, then this output goes Low if the AND Enable is active, and High if it is inactive. Therefore, when data is inserted into one input of the active Low AND Enable gate, it is switched to the output under control of a strobe present on the other AND input. Thus, the decoder performs a demultiplexing function. Note that all unselected outputs are High.

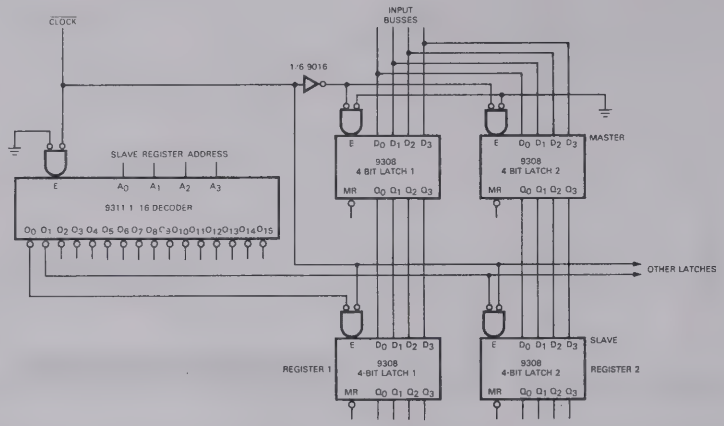
CLOCK DEMULTIPLEXING


CLOCK
ADDRESS
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
OTHER COUNTERS REGISTERS
PE P0 P1 P2 P3
J
CP
K
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
PL
CPU
CPD
MR
9360/9366
TCU
TCD
UP/DOWN COUNTER

Many applications of this demultiplexing principle are possible; that above shows the 9311 decoder serving as a clock demultiplexer. Under control of the address, the clock is routed to the appropriate register or counter. If the address to the decoder changes after the Low-to-High clock transition, there are no glitches or spikes on unselected outputs.

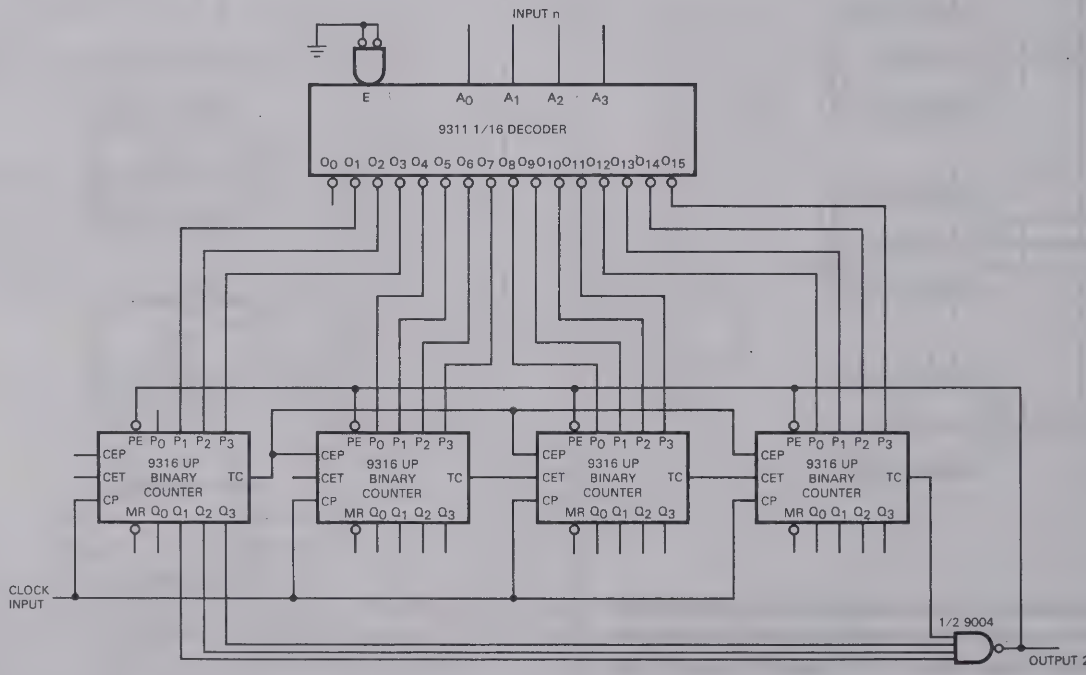
9311 1-OF-16 DECODER APPLICATIONS

SINGLE-MASTER/MULTIPLE-SLAVE FLIP FLOP

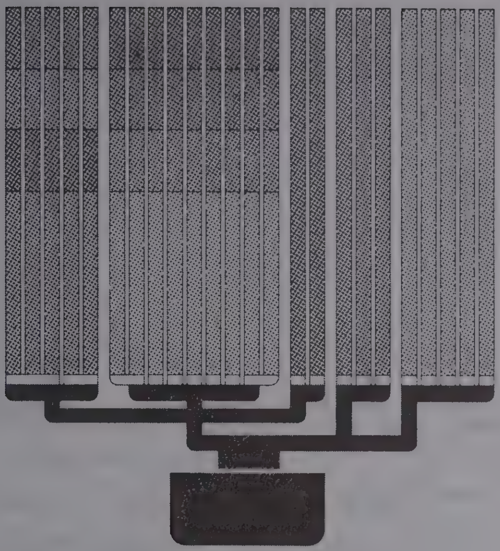

CLOCK
INPUT BUSSES
1/6 9016
SLAVE REGISTER ADDRESS
E
A0 A1 A2 A3
9311 1 16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
E D0 D1 D2 D3
9308
4 BIT LATCH 1
MR Q0 Q1 Q2 Q3
9308
4 BIT LATCH 2
MASTER
OTHER LATCHES
REGISTER 1
9308
4-BIT LATCH 1
9308
4-BIT LATCH 2
SLAVE
REGISTER 2

This application uses the 9311 decoder to control a group of latches working as a single-master/multiple-slave flip flop. When the clock is High, information from a parallel set of busses is entered into the master latches. When the clock line is Low, this information is transferred to the appropriate slave latch addressed by the decoder. Two dual latch blocks function as masters to share the load to drive the 16 slave registers. Alternatively, inverting buffers could be placed between the master and slave for loading purposes but this would cause additional delay. Since each decoder output can drive 10 latch blocks, the circuit shown can be extended to 40-bit words. This multiple slave method offers considerable hardware savings compared to a register approach in which each register contains its own individual master/slave latches.

COUNTER


INPUT n
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK INPUT
1/2 9004
OUTPUT 2ⁿ

A programmable counter can be designed using a 9311 decoder that counts in modulo 2^n, where n is the programmable input. Shown above is a 9311 decoder and four 9316 binary counters capable of counting up to 2^{15}. The input n drives the selected output Low so that when a parallel load occurs, all Highs are written into the register except at the stage represented by the address n. The counter counts pulses and reaches the condition 0000111111111, at which point, the terminal count of the last stage goes High. Ater 14 additional pulses bring the total to 2^n-1, the three remaining inputs to the 9004 gate are High, and the next clock pulse reloads the counter to its original condition. The circuit therefore performs as a 2^n programmable divider.

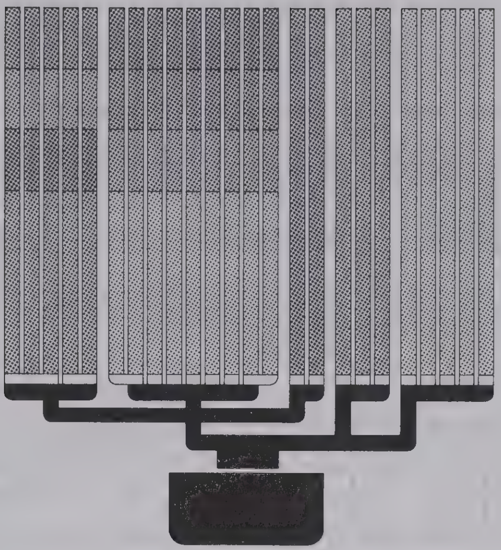


Digital Display Systems

Display Devices
Decoder/Drivers
Decode Formats
Zero Suppression
Multiplexed Systems
Incandescent Display Systems
Fluorescent Display Systems
Gas Discharge Display Systems
Light Emitting Diode Display Systems

DIGITAL DISPLAY SYSTEM SELECTION GUIDE

	Display	Manufacturer	Fairchild Circuit	Type Of Decoder	Active	Outputs Volts	mA	Remarks
Gas Discharge 1-of-10 (Nixie* Type) (All cold cathode)	ZM 1000	Amperex — Slaytorsville, R.I.	**9315** **(7441)**	1-of-10	Low	0—65	7	Direct Drive or Multiplexed
		Burroughs — Plainfield, N.J.	**93141** **(54141/ 74141)**	1-of-10	Low	55	7	All Drive Directly
	NL5750, NL 940, etc.	National Electronics — Japan						
		Raytheon — Quincy, MA.						
Gas Discharge 7-Segment	Panaplex	Burroughs — Plainfield, N.J.	9307	7-Segment	High	Logic Levels 10 Units Loads		Requires Discrete Transistor Interface Direct Drive or Multiplexed
	SP Series	Sperry — Scotsdale, AZ.						
		National Electronics — Japan						
	Elfin	Alco — Lawrence, MA.						
Incandescent 1-of-10 (Projection Displays)			9301	1-of-10	Low	Logic Levels 10 Unit Loads (16 mA Sink)		
			9302	1-of-10	Low	7	16	— Open Collector
			9345 (7445)	1-of-10	Low	30	80	— Open Collector
			93145 (74145)			15	80	— Open Collector
Incandescent 7-Segment		Dialight — Brrolyn, N.Y.	**9317 B**	7-Segment	Low	20	40	All Drive Directly
	Appollo	Diametrics — Torrance, CA.	**C**	7-Segment	Low	30	20	
		EDP — Orlando, FL.	**9357A** **(5446/ 7446)**	7-Segment	Low	30	20	
		James Electronics — Chicago, IL.						
		Luminetics — Pompano Beach, FL.						
		Mesa — Bristol, PA.	**9357B** **(5447/ 7447)**			15	20	
		Pinlight — Fairfield, N.J.						
	Numitron	RCA — Harrison, N.J.	**9370**	7-Segment	Low	7	25	— Contains Input Latches
	Minitron	Readouts — Del Mar, CA.						
LED Common Cathode 5 to 15 mA/ Segment	FND 10	Fairchild Optoelectronics — Palo Alto, CA.	9307	7-Segment	High			Requires Current Limiting Resistors
	MAN 3	Monsanto — Cupertino, CA.	9368	7-Segment	High	1.7	19	— Contains Input Latches
	MOR 33	Motorola — Phoenix, AZ.						
	HP 5082	Hewlett-Packard — Palo Alto, CA.	9369	7-Segment	High	3.5	50	—Multiplex Applications Contains Input Latches
	7200 Series	Hewlett-Packard — Palo Alto, CA.						
15 - 20 mA/ Segment	R7M Elite	Bowmar — Canada	9368	7-Segment	High	1.7	19	— Requires Current Limiting Resistors Contains Input Latches
	FND 70	Fairchild Optoelectronics — Palo Alto, CA.	**9368**	7-Segment	High	1.7	19	— Drives Directly
LED Common Anode	Data Lite 6	Litronix — Cupertino, CA.	9317B	7-Segment	Low	20	40	Requires Current Limiting Resistors
	Data Lite 8		C	7-Segment	Low	30	20	
	Data Lite 10							
	MAN 1	Monsanto — Cupertino, CA.	9357A (5446/ 7446)	7-Segment	Low	30	20	
	MAN 1001							
	SLA 1 - 4	Opcoa — Edison, N.J.	9357B (5447/ 7447)			15	20	
	TIXL 301	Texas Instruments — Dallas, TX.						
	TIXL 302		9370	7-Segment	Low	7	25	— Contains Input Latches
	7730 Series	Hewlett-Packard — Palo Alto, CA.						
Fluorescent	GE Y1938; Y4075; Y4102,05	General Electric — Owensboro, KY.	9307	7-Segment	High			Requires Either Discrete Transistors or Interface Units
	Legitron DG10, 12H, 19C	Legi Electronics — L.A., CA. (Iseden — Japan)						
	NEC DG12E/ LD915,DG10E/ LD938	Nippon Electric — Japan (L.A., CA.)						
	Digivac	Tung-Sol — Livingston, N.Y.						
	Fluorotron	Sylvania — Seneca Falls, N.Y.						



DIGITAL DISPLAY SYSTEMS

INTRODUCTION

Automation continues to demand better display readout devices for point of sale equipment, DVMs, frequency meters, counters, etc. The result is a wide variety of shapes, sizes, input drive requirements and principles of operation of display units. Most electronic readout devices use neon, fluorescent, incandescent, electroluminescent or gallium arsenide methods to form or illuminate the readout display.

A display decoder/driver takes 4-bit (usually BCD) input data and decodes it into the correct format for the particular display being activated. Outputs must be of sufficient current and voltage and correct polarity to drive the display. Such decoder/drivers are now available for many display types in a single integrated circuit.

DISPLAYS

One of the oldest electronic numeric readouts is the one-of-ten display such as the NIXIE* tube. An inherent disadvantage is that each of the numbers within the tube is not on the same plane. This is very evident when a number of displays are used side by side. Additionally, the red illumination of the displays makes it difficult to change the readout color. Also, they are difficult to multiplex because of relatively high voltage requirements. Seven-segment displays have become popular due to their lower prices and pleasant, modern numeral format. These displays are available in a wide variety of size, color and type.

Incandescent displays can be made in a wide range of sizes and colors and are among the brightest available depending upon the lamps used. Until recently their main disadvantage was reliability due to segment failure. New materials, packages and methods however, have improved their reliability.

Many newer incandescent displays have all seven segment filaments contained within a single vacuum envelope and are compatible with standard DTL and TTL voltages. Multiplexing incandescent readouts doesn't offer much advantage in part count as each of the display segments requires a diode to stop sneak electrical paths.

Cold cathode displays, also known as *neon, gas discharge,* or *plasma* displays are improved nixie-type displays with seven segments instead of 10 numeral cathodes. Easily read and red-orange in color, they are available in sizes up to 0.75 inches high. They do have a disadvantage in that a high anode potential is required making them difficult to multiplex.

Fluorescent displays are blue-green, available to approximately 0.6 inch character height, and are used primarily in imported calculators. Their relatively low current and voltage requirements make them easy to multiplex.

The *light emitting diode* is a modern technology, solid state device using either gallium arsenide or gallium arsenide phosphide. Generally, the advantage of these displays lies in their smaller size, more reliable operation under severe mechanical conditions, and voltage current compatibility with standard integrated circuit technology. LEDs are available from 0.1 inch to 0.8 inch heights and are typically red in color; however, yellows and greens are offered at a price. Most of the smaller 0.1 inch LEDs are used in domestic hand calculators.

Liquid crystal displays are unique because they scatter, rather than generate, light. There are two basic types: reflective which requires front illumination, and transmissive which requires rear illumination. Liquid crystal display devices have the lowest power requirements of any display; however, they require an ac drive system which makes them difficult to multiplex. Short operating life, low reliability, and sensitivity to ultraviolet light have impeded the progress of these displays.

Other technologies for making 7-segment displays are electroluminescent and light emitting thin films with very high voltage requirements, but IC decoder/drivers for these displays are cumbersome and difficult to make.

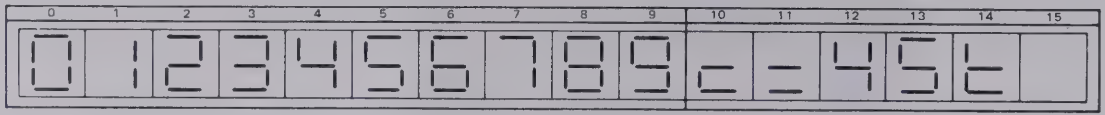
---

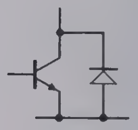
*NIXIE is a registered trademark of Burroughs Corporation

DIGITAL DISPLAY SYSTEMS

7-SEGMENT DECODER/DRIVERS

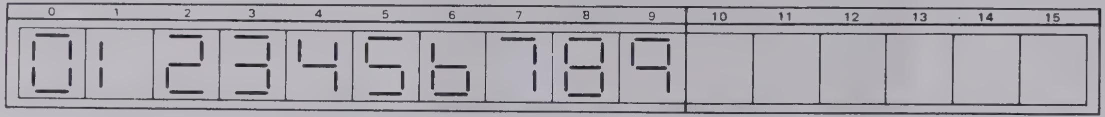
9307



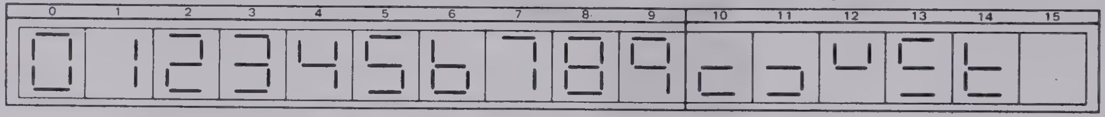


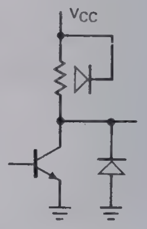
OPEN COLLECTOR OUTPUT CIRCUIT

9317



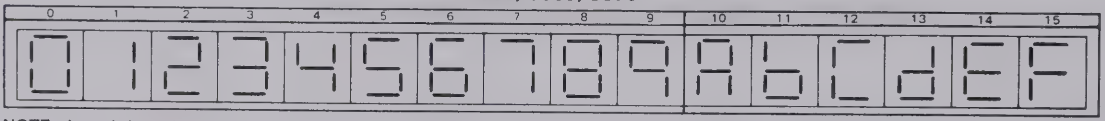
9357 (5446/7446), 9358 (5448/7448), 9359 (5449/7449)



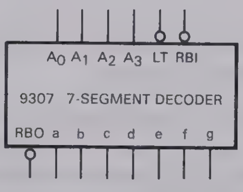


RESISTIVE PULL-UP OUTPUT CIRCUIT

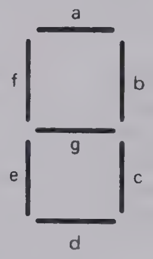
9368, 9369, 9370



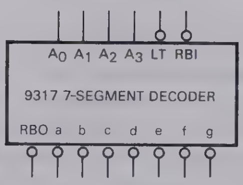
NOTE: b and d are lower case.



SEGMENTS ACTIVE HIGH RESISTIVE PULL-UP OUTPUT CIRCUIT



SEGMENT DESIGNATION



SEGMENTS ACTIVE LOW OPEN COLLECTOR OUTPUT CIRCUIT

		INPUTS				OUTPUTS								DECIMAL OR FUNCTION
$\overline{LT}$	$\overline{RBI}$	A_0	A_1	A_2	A_3	a	b	c	d	e	f	g	$\overline{RBO}$	
L	X	X	X	X	X	H	H	H	H	H	H	H	H	TEST
H	L	L	L	L	L	L	L	L	L	L	L	L	L	BLANK
H	H	L	L	L	L	H	H	H	H	H	H	L	H	0
↓	X	H	L	L	L	L	H	H	L	L	L	L	H	1
↓	↓	L	H	L	L	H	H	L	H	H	L	H	H	2
↓	↓	H	H	L	L	H	H	H	H	L	L	H	H	3
↓	↓	L	L	H	L	L	H	H	L	L	H	H	H	4
↓	↓	H	L	H	L	H	L	H	H	L	H	H	H	5
↓	↓	L	H	H	L	H	L	H	H	H	H	H	H	6
↓	↓	H	H	H	L	H	H	H	L	L	L	L	H	7
↓	↓	L	L	L	H	H	H	H	H	H	H	H	H	8
↓	↓	H	L	L	H	H	H	H	H	L	H	H	H	9
↓	↓	L	H	L	H	L	L	L	H	H	L	H	H	10
↓	↓	H	H	L	H	L	L	L	H	L	L	H	H	11
↓	↓	L	L	H	H	L	H	H	L	L	H	H	H	12
↓	↓	H	L	H	H	H	L	H	H	L	H	H	H	13
↓	↓	L	H	H	H	L	L	L	H	H	H	H	H	14
H	X	H	H	H	H	L	L	L	L	L	L	L	H	15

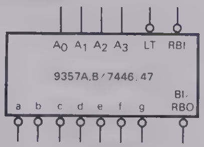
		INPUTS				OUTPUTS								DECIMAL OR FUNCTION
$\overline{LT}$	$\overline{RBI}$	A_0	A_1	A_2	A_3	$\overline{a}$	$\overline{b}$	$\overline{c}$	$\overline{d}$	$\overline{e}$	$\overline{f}$	$\overline{g}$	$\overline{RBO}$	
L	X	X	X	X	X	L	L	L	L	L	L	L	H	TEST
H	L	L	L	L	L	H	H	H	H	H	H	H	L	BLANK
H	H	L	L	L	L	L	L	L	L	L	L	H	H	0
↓	X	H	L	L	L	H	H	H	H	L	L	H	H	1
↓	↓	L	H	L	L	L	L	H	L	L	H	L	H	2
↓	↓	H	H	L	L	L	L	L	L	H	H	L	H	3
↓	↓	L	L	H	L	H	L	L	H	H	L	L	H	4
↓	↓	H	L	H	L	L	H	L	L	H	L	L	H	5
↓	↓	L	H	H	L	H	H	L	L	L	L	L	H	6
↓	↓	H	H	H	L	L	L	L	H	H	H	H	H	7
↓	↓	L	L	L	H	L	L	L	L	L	L	L	H	8
↓	↓	H	L	L	H	L	L	L	H	H	L	L	H	9
↓	↓	L	H	L	H	H	H	H	H	H	H	H	L	10
↓	↓	H	H	L	H	H	H	H	H	H	H	H	L	11
↓	↓	L	L	H	H	H	H	H	H	H	H	H	L	12
↓	↓	H	L	H	H	H	H	H	H	H	H	H	L	13
↓	↓	L	H	H	H	H	H	H	H	H	H	H	L	14
H	X	H	H	H	H	H	H	H	H	H	H	H	L	15

Decoder/drivers for 7-segment displays accept a 4-bit BCD 8421 input code and produce the appropriate outputs for selection of segments in a 7-segment matrix display used for representing the decimal numbers 0—9 and alpha characters when necessary. The seven outputs (a, b, c, d, e, f, g) of the decoder select the corresponding segments in the matrices shown. The numeric designations chosen to represent the decimal numbers are shown in the decode fonts above.

The 9368, 9369, and 9370 devices have built in latch circuits for data storage. Latches are activated by a single active Low Enable input. When the Enable is Low, the latches are transparent; when High, the data present at the inputs prior to the Enable going High is stored.

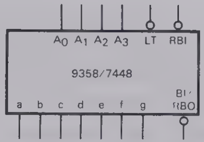
When decoders have active Low input Lamp Test LT, this input overrides all other conditions and enables a check on possible display malfunctions. The $\overline{RBO}$ terminal of the decoder can be OR-tied with a modulating signal via isolating buffer for either pulse duration intensity modulations or display blanking.

7-SEGMENT DECODER/DRIVERS


A0 A1 A2 A3 LT RBI
9357A,B/7446, 47
BI·RBO
a b c d e f g

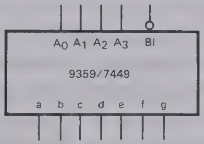
SEGMENTS ACTIVE LOW
OPEN COLLECTOR OUTPUT CIRCUIT

INPUTS						OUTPUTS							$\overline{BI}$ OR $\overline{RBO}$	DECIMAL OR FUNCTION
$\overline{LT}$	$\overline{RBI}$	A_0	A_1	A_2	A_3	$\overline{a}$	$\overline{b}$	$\overline{c}$	$\overline{d}$	$\overline{e}$	$\overline{f}$	$\overline{g}$		
L	X	X	X	X	X	L	L	L	L	L	L	L	H	TEST
H	L	L	L	L	L	H	H	H	H	H	H	H	L	BLANK
H	H	L	L	L	L	L	L	L	L	L	L	H	H	0
	X	H	L	L	L	H	L	L	H	H	H	H	H	1
		L	H	L	L	L	L	H	L	L	H	L	H	2
		H	H	L	L	L	L	L	L	H	H	L	H	3
		L	L	H	L	H	L	L	H	H	L	L	H	4
		H	L	H	L	L	H	L	L	H	L	L	H	5
		L	H	H	L	H	H	L	L	L	L	L	H	6
		H	H	H	L	L	L	L	H	H	H	H	H	7
		L	L	L	H	L	L	L	L	L	L	L	H	8
		H	L	L	H	L	L	L	H	H	L	L	H	9
		L	H	L	H	H	H	H	L	L	H	L	H	10
		H	H	L	H	H	H	L	L	H	H	L	H	11
		L	L	H	H	H	L	H	H	H	L	L	H	12
		H	L	H	H	L	H	H	L	H	L	L	H	13
		L	H	H	H	H	H	H	L	L	L	L	H	14
H	X	H	H	H	H	H	H	H	H	H	H	H	H	15


A0 A1 A2 A3 LT RBI
9358/7448
BI·RBO
a b c d e f g

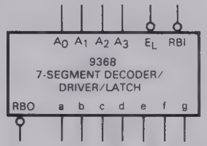
SEGMENTS ACTIVE HIGH
RESISTIVE PULL-UP OUTPUT CIRCUIT

INPUTS						OUTPUTS							$\overline{BI}$ OR $\overline{RBO}$	DECIMAL OR FUNCTION
$\overline{LT}$	$\overline{RBI}$	A_0	A_1	A_2	A_3	a	b	c	d	e	f	g		
L	X	X	X	X	X	H	H	H	H	H	H	H	H	TEST
H	L	L	L	L	L	L	L	L	L	L	L	L	L	BLANK
H	H	L	L	L	L	H	H	H	H	H	H	L	H	0
	X	H	L	L	L	L	H	H	L	L	L	L	H	1
		L	H	L	L	H	H	L	H	H	L	H	H	2
		H	H	L	L	H	H	H	H	L	L	H	H	3
		L	L	H	L	L	H	H	L	L	H	H	H	4
		H	L	H	L	H	L	H	H	L	H	H	H	5
		L	H	H	L	L	L	H	H	H	H	H	H	6
		H	H	H	L	H	H	H	L	L	L	L	H	7
		L	L	L	H	H	H	H	H	H	H	H	H	8
		H	L	L	H	H	H	H	L	L	H	H	H	9
		L	H	L	H	L	L	L	H	H	L	H	H	10
		H	H	L	H	L	L	H	H	L	L	H	H	11
		L	L	H	H	L	H	L	L	L	H	H	H	12
		H	L	H	H	H	L	L	H	L	H	H	H	13
		L	H	H	H	L	L	L	H	H	H	H	H	14
H	X	H	H	H	H	L	L	L	L	L	L	L	L	15

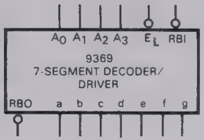

A0 A1 A2 A3 BI
9359/7449
a b c d e f g

SEGMENTS ACTIVE HIGH
OPEN COLLECTOR OUTPUT CIRCUIT

INPUTS					OUTPUTS							DECIMAL OR FUNCTION
$\overline{BI}$	A_0	A_1	A_2	A_3	a	b	c	d	e	f	g	
L	X	X	X	X	L	L	L	L	L	L	L	BLANK
H	L	L	L	L	H	H	H	H	H	H	L	0
	H	L	L	L	L	H	H	L	L	L	L	1
	L	H	L	L	H	H	L	H	H	L	H	2
	H	H	L	L	H	H	H	H	L	L	H	3
	L	L	H	L	L	H	H	L	L	H	H	4
	H	L	H	L	H	L	H	H	L	H	H	5
	L	H	H	L	L	L	H	H	H	H	H	6
	H	H	H	L	H	H	H	L	L	L	L	7
	L	L	L	H	H	H	H	H	H	H	H	8
	H	L	L	H	H	H	H	L	L	H	H	9
	L	H	L	H	L	L	L	H	H	L	H	10
	H	H	L	H	L	L	H	H	L	L	H	11
	L	L	H	H	L	H	L	L	L	H	H	12
	H	L	H	H	H	L	L	H	L	H	H	13
	L	H	H	H	L	L	L	H	H	H	H	14
H	H	H	H	H	L	L	L	L	L	L	L	15


A0 A1 A2 A3 EL RBI
9368
7-SEGMENT DECODER/
DRIVER/LATCH
RBO a b c d e f g

SEGMENTS ACTIVE HIGH
CURRENT LIMITED OPEN EMITTER OUTPUT CIRCUIT


A0 A1 A2 A3 EL RBI
9369
7-SEGMENT DECODER/
DRIVER
RBO a b c d e f g

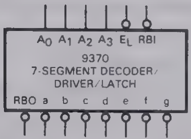
SEGMENTS ACTIVE HIGH
OPEN EMITTER OUTPUT CIRCUIT

LATCH ENABLE	OUTPUTS
L	Data follows input
H	Store

9368/9369

	INPUTS					OUTPUTS								
	$\overline{RBI}$	A_3	A_2	A_1	A_0	a	b	c	d	e	f	g	$\overline{RBO}$	DISPLAYS
*	X	X	X	X	X	L	L	L	L	L	L	L	L*	BLANK
0	L	L	L	L	L	L	L	L	L	L	L	L	L	BLANK
0	H	L	L	L	L	H	H	H	H	H	H	L	H	0
1	X	L	L	L	H	L	H	H	L	L	L	L		1
2		L	L	H	L	H	H	L	H	H	L	H		2
3		L	L	H	H	H	H	H	H	L	L	H		3
4		L	H	L	L	L	H	H	L	L	H	H		4
5		L	H	L	H	H	L	H	H	L	H	H		5
6		L	H	H	L	H	L	H	H	H	H	H		6
7		L	H	H	H	H	H	H	L	L	L	L		7
8		H	L	L	L	H	H	H	H	H	H	H		8
9		H	L	L	H	H	H	H	L	L	H	H		9
10		H	L	H	L	H	H	H	L	H	H	H		A
11		H	L	H	H	L	L	H	H	H	H	H		b
12		H	H	L	L	H	L	L	H	H	H	H		C
13		H	H	L	H	L	H	H	H	H	L	H		d
14		H	H	H	L	H	L	L	H	H	H	H		E
15	X	H	H	H	H	H	L	L	L	L	H	H	H	F

*RBO used as an input


A0 A1 A2 A3 EL RBI
9370
7-SEGMENT DECODER/
DRIVER/LATCH
RBO a b c d e f g

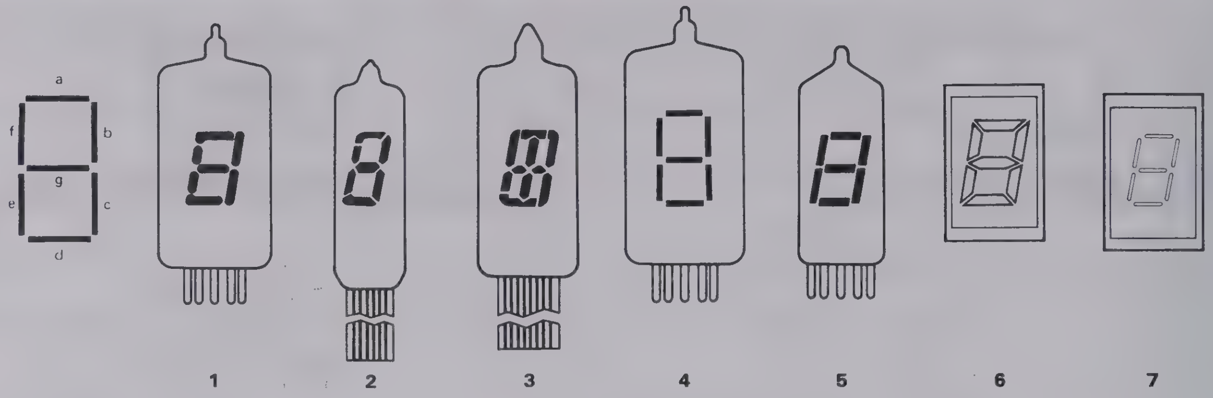
LATCH ENABLE	OUTPUTS
L	Data follows input
H	Store

SEGMENTS ACTIVE LOW
OPEN COLLECTOR OUTPUT CIRCUIT

	INPUTS					OUTPUTS								
	$\overline{RBI}$	A_3	A_2	A_1	A_0	$\overline{a}$	$\overline{b}$	$\overline{c}$	$\overline{d}$	$\overline{e}$	$\overline{f}$	$\overline{g}$	$\overline{RBO}$	DISPLAY
*	X	X	X	X	X	H	H	H	H	H	H	H	L*	BLANK
0	L	L	L	L	L	H	H	H	H	H	H	H	L	BLANK
0	H	L	L	L	L	L	L	L	L	L	L	H	H	0
1	X	L	L	L	H	H	L	L	H	H	H	H		1
2		L	L	H	L	L	L	H	L	L	H	L		2
3		L	L	H	H	L	L	L	L	H	H	L		3
3		L	H	L	L	H	L	L	H	H	L	L		4
5		L	H	L	H	L	H	L	L	H	L	L		5
6		L	H	H	L	L	H	L	L	L	L	L		6
7		L	H	H	H	L	L	L	H	H	H	H		7
8		H	L	L	L	L	L	L	L	L	L	L		8
9		H	L	L	H	L	L	L	H	H	L	L		9
10		H	L	H	L	L	L	L	H	L	L	L		A
11		H	L	H	H	H	H	L	L	L	L	L		b
12		H	H	L	L	L	H	H	L	L	L	L		C
13		H	H	L	H	H	L	L	L	L	H	L		d
14		H	H	H	L	L	H	H	L	L	L	L		E
15	X	H	H	H	H	L	H	H	H	L	L	L	H	F

*RBO used as an input

7-SEGMENT DECODE FORMAT



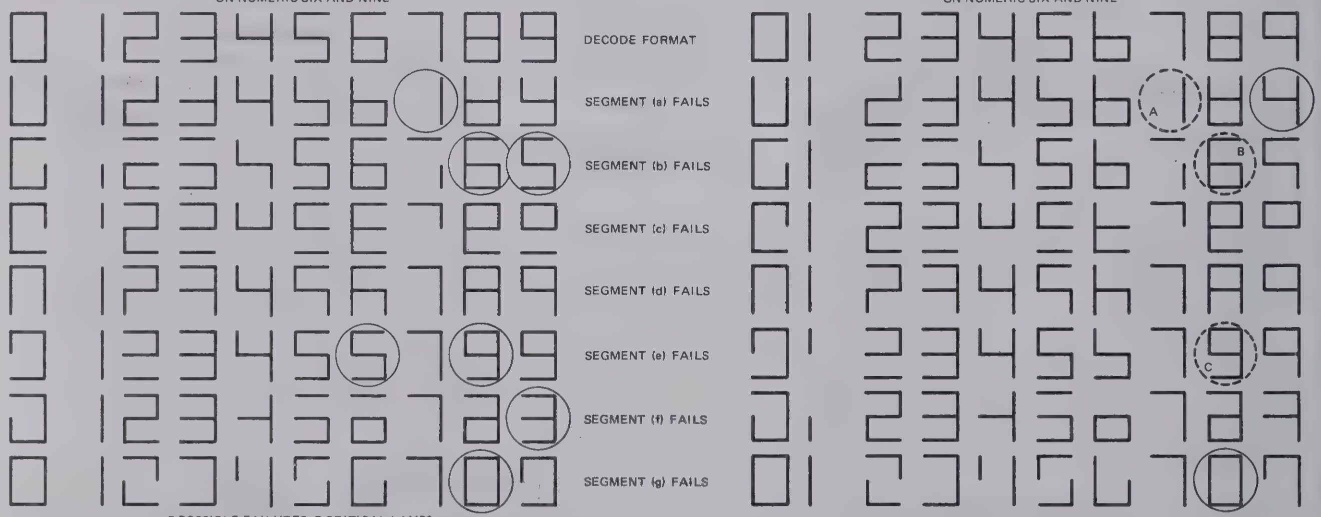
Decoder/driver logic for 7-segment displays is based on four basic decode formats. Output drive capability and polarity are adjusted to suit the display device. The decoder/drivers listed are grouped by logic decode format.

- 9307
- 9317
- 9357 (7445 & 7446), 9358 (7448), 9359 (7449)
- 9368, 9369, 9370

Decode format is an important consideration for 7-segment displays, not only for appearance and reliability, but also because certain segment shapes selected by manufacturers do not represent certain numerals well. For example, a left-hand one or a six without a tail would be difficult to decipher using the displays in 1, 2, and 3 above.

SEVEN SEGMENT DECODER/DRIVER WITH NUMERIC ONE ON THE RIGHT HAND SIDE AND TAILS ON NUMERIC SIX AND NINE

9317 DECODER/DRIVER WITH NUMERIC ONE ON THE LEFT HAND SIDE AND NO TAILS ON NUMERIC SIX AND NINE



7 POSSIBLE FAILURES, 5 CRITICAL LAMPS

ONLY 5 POSSIBLE FAILURES, 4 CRITICAL LAMPS

CONSIDERING A, B & C FAILURES FOR THE 9317 FORMAT ON THE RIGHT ABOVE, THERE IS A READABLE NUMBER. FAILURES A ①, B ⑥, AND C ⑨ ARE IN THE WRONG DECODE FONT. A CAREFUL OBSERVER WOULD NOTICE THAT THESE ARE THE WRONG NUMBERS. OF COURSE, THEY SHOULD READ 7 FOR A, 8 FOR B, AND 8 FOR C.

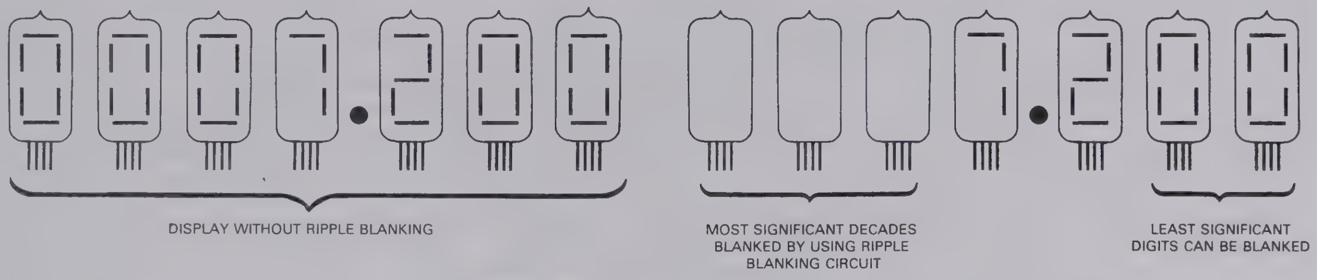
Another consideration involves possible failures and critical lamp considerations for both formats (decimal one on right segments with tails on six and nine and decimal one on left segments without tails on six and nine). Note that for the 9317 font, dotted circles show actual numerals even with segment failures but the numeric formats are wrong. Thus, there are actually only two possible failures and two critical lamps if the viewer realizes that the numbers 1, 6 and 9 are in the wrong format. Considering all readable numerals, there are five failures and four critical lamps.

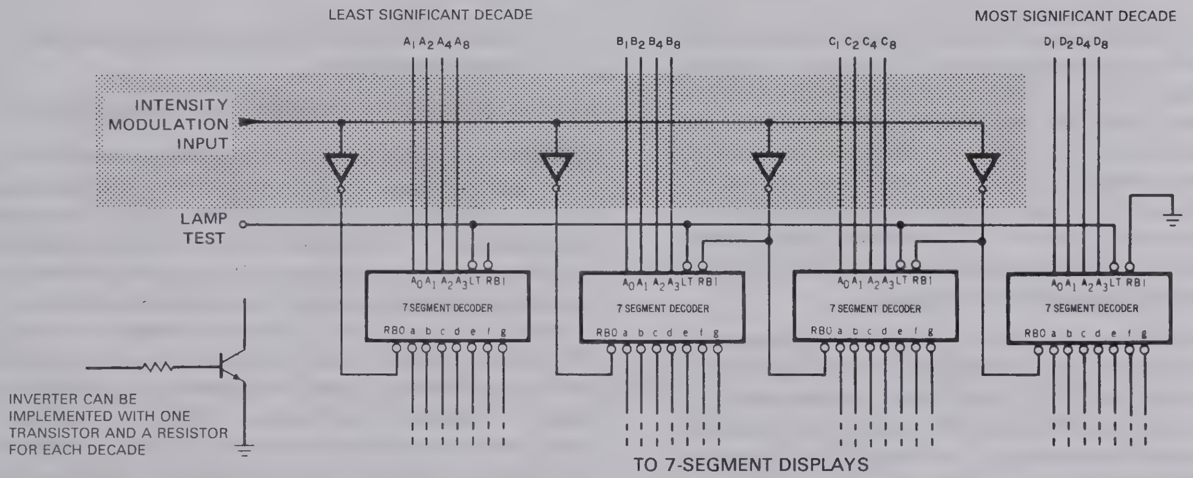
The elimination of the tails on the six and nine provides the greatest single contribution to reliability. Placing the decimal one on the left hand segments also increases reliability. This is because the distribution of segment use for all ten numerals is more equal.

LAMP SEGMENT USED	SEGMENT USE DISTRIBUTION NUMERIC ONE POSITION ON	
	L.H.SIDE	R.H.SIDE
B	7	8
C	8	9
E	5	4
F	7	6

Because of improved segment use distribution and fewer critical lamps, the 9317 format is an advantage in military or other applications where there is increased concern with erroneous readout.

ZERO SUPPRESSION


DISPLAY WITHOUT RIPPLE BLANKING
MOST SIGNIFICANT DECADES BLANKED BY USING RIPPLE BLANKING CIRCUIT
LEAST SIGNIFICANT DIGITS CAN BE BLANKED


LEAST SIGNIFICANT DECADE
MOST SIGNIFICANT DECADE
A1 A2 A4 A8
B1 B2 B4 B8
C1 C2 C4 C8
D1 D2 D4 D8
INTENSITY MODULATION INPUT
LAMP TEST
A0 A1 A2 A3 LT RBI
7 SEGMENT DECODER
RBO a b c d e f g
INVERTER CAN BE IMPLEMENTED WITH ONE TRANSISTOR AND A RESISTOR FOR EACH DECADE
TO 7-SEGMENT DISPLAYS

ALL INVERTERS ARE DTL 9936 OR OPEN COLLECTOR TTL 9N05 (7405)

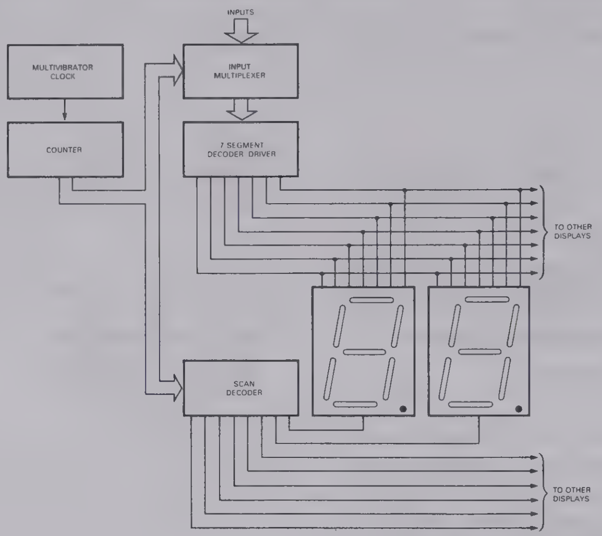
Leading or trailing edge zero suppression is particularly valuable in multidigit displays with a decimal point. Legibility is enhanced by blanking irrelevant decades. All devices except the 9359 have provision for automatic blanking of the leading and/or trailing edge zeros in a multidigit decimal number, resulting in an easily readable decimal display, conforming to normal writing practice. In an 8-digit, mixed integer, fraction decimal representation, using the automatic blanking capability, 0007.200 would be displayed as 7.2. Leading edge zero suppression is obtained by connecting the Ripple Blanking Output RBO of a decoder to the Ripple Blanking Input RBI of the next lower stage device. The most significant decoder stage should have the RBI input grounded. Since the suppression of the least significant integer zero in a number is not usually desired, the RBI input of this decoder stage should be left open. A similar procedure for the fractional part of a display provides automatic suppression of trailing edge zeros.

The 9368, 9369, and 9370 devices have built in latch circuits for data storage. Latches are activated by a single active Low Enable input. When the Enable is Low, the latches are transparent; when High, the data present at the inputs prior to the Enable going High is stored. Note that with the 9368, 9369 and 9370, the RBI and RBO is activated by data stored within their latch functions.

As shown, the RBI of the most significant decade decoder is grounded. If a BCD zero is present on this decade's inputs, this display is blanked and RBO is Low. The remainder of the decoders operate in the same manner; consequently, this system is called Ripple Blanking.

The RBO of these decoders has two functions: In addition to its use as an active Low output to detect BCD zeros when RBI is Low, it can also be used as an OR-tied input for either blanking the entire display or controlling brightness by applying pulse duration intensity modulation. The OR-tie capability of the RBO is tied to open collector TTL, DTL or discrete device inverters to control brightness shown in shaded area.

DISPLAY MULTIPLEXING SYSTEMS



Typically, multiplexing is a result of economic considerations such as requirements for simple drive circuitry, or minimum interconnections between the displays and the system providing the data to be displayed. The multiplexed or direct drive display selection is based upon each system's design characteristics. Overall circuit complexity and cost are primary considerations in the majority of consumer and industrial design efforts as opposed to military applications where the priority is more commonly maximum reliability.

In some systems, this choice is quite clear. For example, the popular pocket calculators use a single chip producing data in a serialized format while multiple digit LED displays employ data bussing for ease of manufacturing and to reduce connections. Therefore, the connections and drive units between display and chip are held to a minimum.

Conversely, in a system where the data is presented in parallel format and close to the display, multiplexing the data would greatly increase system complexity and cost. Any data sent a considerable distance should be sent on a single cable and serialized (an advanced form of multiplexing). However, a display subsystem located a few feet away may warrant a simpler form of multiplexing using fewer wires. Even though purchase price and maintenance costs of cables, connectors and joints is increasing, the cost of MSI per function is decreasing. Multiplexing and the degree of multiplexing, therefore, is to be considered carefully.

Five units are used in this basic multiplexing circuit.

- A decoder/driver usually decoding BCD input data to the code requirements of the display device, normally 7-segment or 1-of-10 decode.
- An input address selector (multiplexer or shift register) taking the BCD input data to be displayed to the decoder/driver.
- A scan decoder selecting the display to be energized.
- A scan counter addressing the scan decoder and input address selector.
- A clock input determining the multiplexing rate.

Multiplexing a small number of displays is not economical unless the data input is in serial form. Multiplexing can offer advantages after about four digits for some display types. The maximum number of digits is about 12 (except with LED displays). Beyond this number, it becomes difficult to power each display unit for its diminishing share of time. For a given brightness the display will require a given power whether or not it is multiplexed. When multiplexed, this power is supplied to the display in pulses for a fraction of the complete scan cycle. Hence, voltage and/or current must be increased over that required for the dc conditions. Multiplexing incandescent displays is less attractive because each filament of the display requires a diode to stop sneak electrical paths.

Advantages:

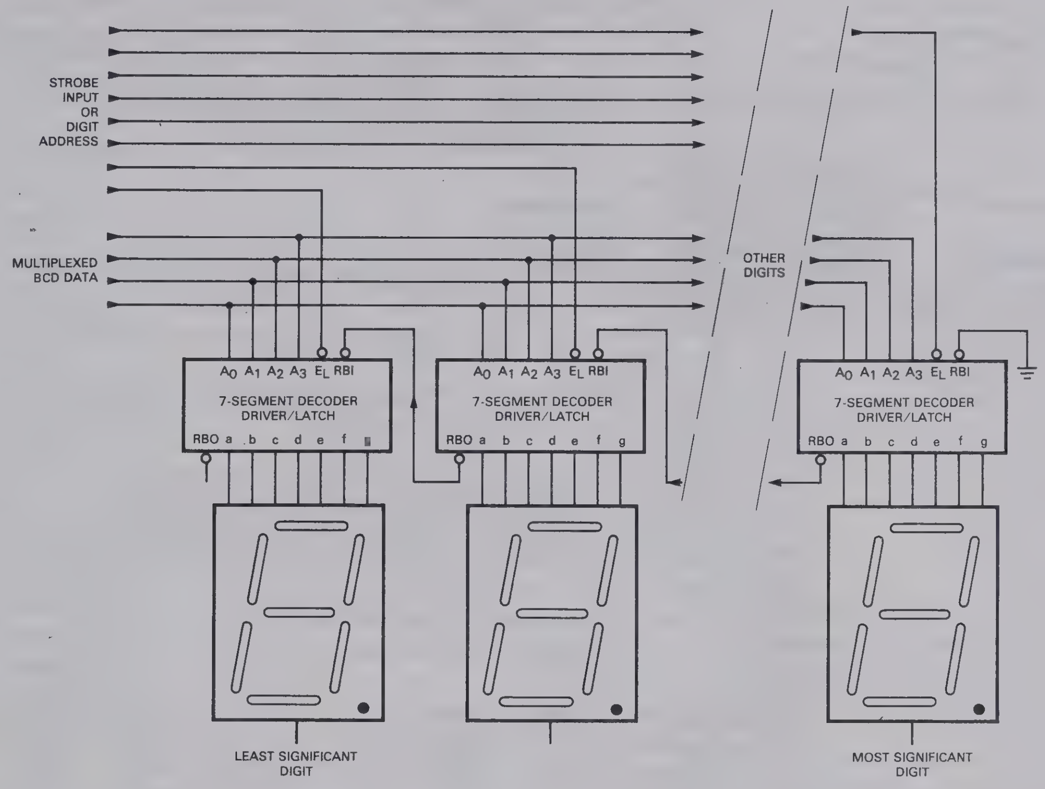
- Fewer parts—lower system cost.
- Easier printed circuit layout.
- Reduced interconnection wiring when readout is remote from display logic.
- System hook-up requires fewer parts.

Disadvantages:

- Higher operating voltages or current required for equivalent brightness.
- Scan rate must be above 1 kHz to reduce flicker.
- Careful uncoupling of power supplies to stop switching transients is required.
- A clock is required to drive multiplexed systems.
- Clock failure results in full power to displays, damaging an unprotected system.

DIGITAL DISPLAY SYSTEMS

DISPLAY DEMULTIPLEXING SYSTEM


STROBE INPUT OR DIGIT ADDRESS
MULTIPLEXED BCD DATA
OTHER DIGITS
A_0 A_1 A_2 A_3 $\overline{E}_L$ RBI
7-SEGMENT DECODER DRIVER/LATCH
RBO a b c d e f g
LEAST SIGNIFICANT DIGIT
MOST SIGNIFICANT DIGIT

9368 Sources 19 mA at 1.7 V, Drives FND 70 Directly
9370 Sinks 25 mA, $V_{CC\ MAX}$ 7 V

This system takes advantage of the storage capabilities of the 7-segment decoder/driver/latch combination. The input data is in multiplexed form while the displays operate in the static mode.

The principle advantage of this system is that 5 V incandescent or FND 70 LED displays are driven directly without any other components using the existing 5 V supply. There are other advantages over the standard multiplex system.

- Low multiplexing rates can be used without flicker.
- Relatively low power line spiking.
- Economically sound on many systems up to 5 digits (incandescents allow even more).
- Clock failure does not cause excessive voltage or currents to be applied to any one display.

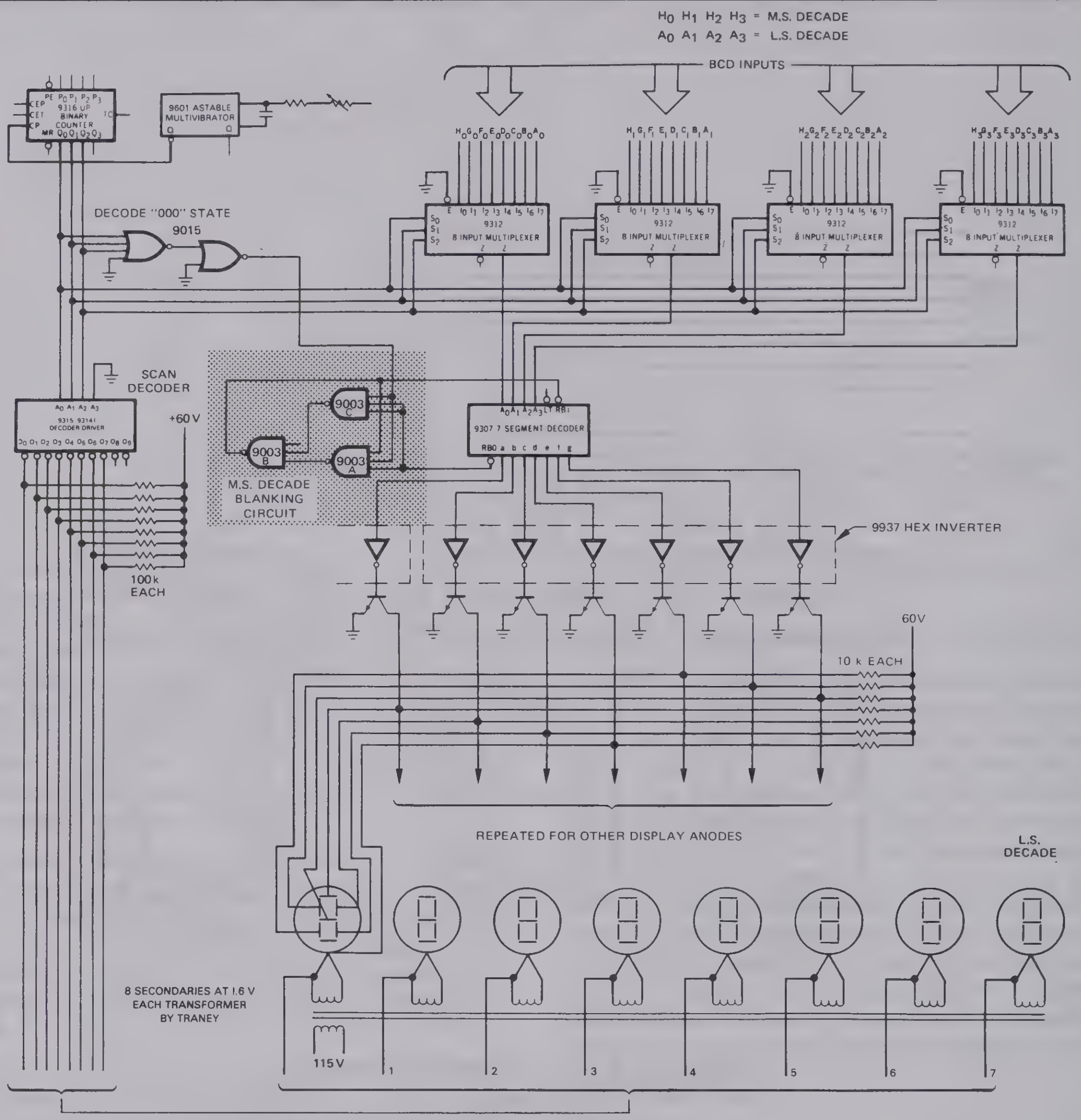
Some disadvantages include:

- BCD data and strobe timing (digit address) is critical.
- PC layout can be slightly more difficult depending on display mounting configuration.

The input BCD data must be stable during the last 30 ns of the active Low enable pulse. All strobe pulse glitches must be avoided.

This method is particularly advantageous in the case of 5 V incandescent displays because the total cost of the decoder/driver/latches compares with the standard multiplex system which uses seven diodes (high insertion cost) per display, two power supply voltages, and a clock failure detection system.

A SIMPLE, 8-DIGIT MULTIPLEXED SYSTEM


H0 H1 H2 H3 = M.S. DECADE
A0 A1 A2 A3 = L.S. DECADE
BCD INPUTS
9316 UP BINARY COUNTER
9601 ASTABLE MULTIVIBRATOR
DECODE "000" STATE
9015
9312 8 INPUT MULTIPLEXER
SCAN DECODER
9315 93141 DECODER DRIVER
+60 V
9003 C
9003 B
9003 A
M.S. DECADE BLANKING CIRCUIT
9307 7 SEGMENT DECODER
9937 HEX INVERTER
100k EACH
60V
10 k EACH
REPEATED FOR OTHER DISPLAY ANODES
L.S. DECADE
8 SECONDARIES AT 1.6 V EACH TRANSFORMER BY TRANEY
115V
1
2
3
4
5
6
7

This circuit illustrates one of the simpler and more straightforward methods of a multiplexing system. Four 9312 8-input multiplexers select the BCD input data for the 9307 decoder. Address for the multiplexers and the scan decoder from the 9316 binary counter uses only the first three bits. Each 3-bit address selects the BCD input data to be displayed and the scan decoder energizes the correct display readout device. The counter progresses through all eight counts and repeats, activating each display unit for 1/8th of the scanning cycle. Higher operating voltage must be applied for the displays to have the same brightness as displays in a static mode (dc). The circuit as shown uses fluorescent diode displays. Other displays may be substituted with the appropriate drive circuitry.

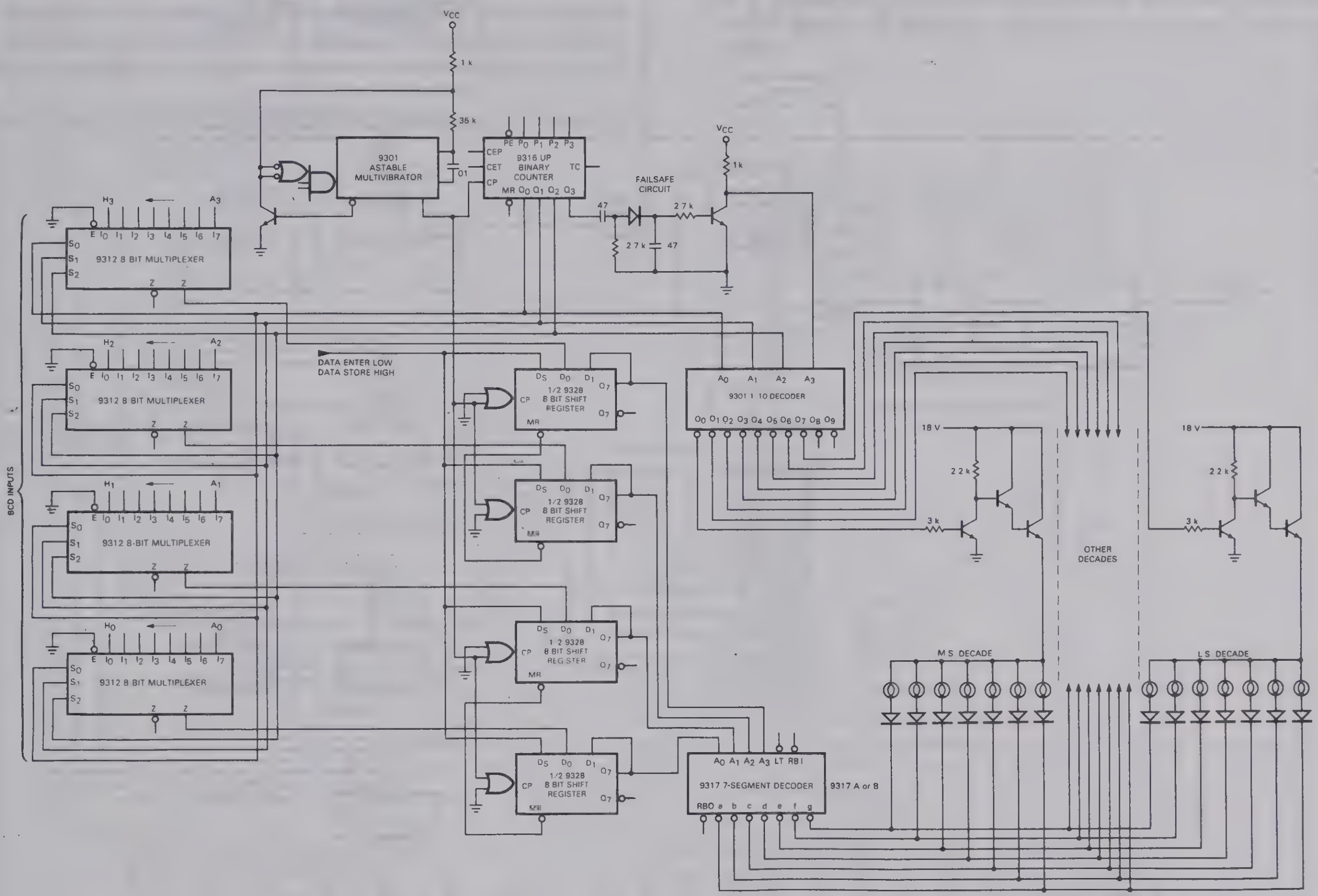
MOST SIGNIFICANT DECADE BLANKING

The objective of the shaded circuit is to hold $\overline{RBI}$ Low during a scan period until a decimal number other than zero appears and then shift $\overline{RBI}$ High for the rest of the scan period. The most significant decade is at the start of the scan period; the least significant is at the end.

When $\overline{RBI}$ is Low, any BCD zeros to the decoder will be blanked on the display, however any BCD number is displayed. Control of this input achieves most significant digit blanking.

The circuit sets the flip-flop (gates A and B) so gate B output is Low at the start of the scan period (zero count on scan decoder, active Low). With $\overline{RBI}$ Low and BCD zeros into the decoder, $\overline{RBO}$ is Low. With the first decimal number of the scan period, $\overline{RBO}$ goes High. As the reset line is also High, gate C output goes Low setting flip-flop, (gate B output High, therefore $\overline{RBI}$ is High). This sequence repeats each scan period.

MULTIPLEXED 8-DIGIT DISPLAY WITH MEMORY

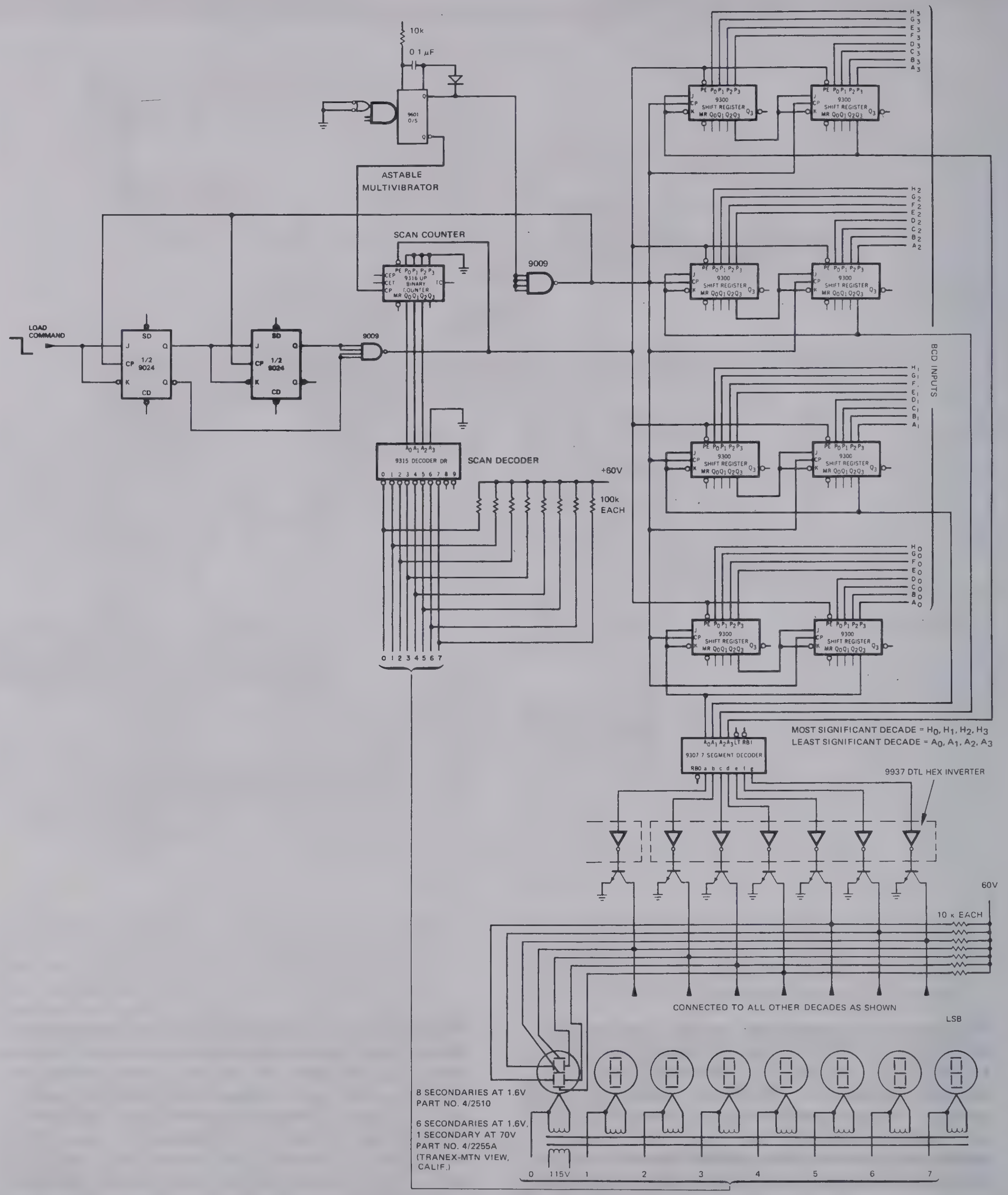

VCC
1 k
35 k
9301 ASTABLE MULTIVIBRATOR
.01
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
FAILSAFE CIRCUIT
47
2.7 k
2.7 k
47
VCC
1 k
H3
A3
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
9312 8 BIT MULTIPLEXER
Z Z
H2
A2
9312 8 BIT MULTIPLEXER
H1
A1
9312 8-BIT MULTIPLEXER
H0
A0
9312 8 BIT MULTIPLEXER
BCD INPUTS
DATA ENTER LOW
DATA STORE HIGH
DS D0 D1
1/2 9328 8 BIT SHIFT REGISTER
CP
MR
Q7
A0 A1 A2 A3
9301 1:10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
18 V
2.2 k
3 k
OTHER DECADES
18 V
2.2 k
3 k
M.S. DECADE
L.S. DECADE
A0 A1 A2 A3 LT RBI
9317 7-SEGMENT DECODER
9317 A or B
RBO a b c d e f g

ALL TRANSISTORS 2N3567
ALL DIODES 1N333
ALL LAMPS GE381 6 V, 20 mA
OR NEW 5 V INCANDESCENT

This multiplexing system has storage capability. Each bit of the 4-bit BCD code input is supplied to one of the four 9312 multiplexers. The multiplexer input addresses are selected by a 3-bit binary output from 9316 counter. The multiplexer outputs are fed into four 8-bit shift registers when Data Select D_S is Low. The multiplexer steps from one input address to the next on each clock pulse. Simultaneously, information is propagated through the shift registers to the decoder/driver. With D_S High, shift register output information is returned to the input to form a recirculating memory. Information continues to circulate as long as D_S is High and power is applied. Information must be present at the inputs for at least one complete scan period to insure that the multiplexers sample all inputs.

The failsafe circuit detects clock failure and disables the address to avoid excessive voltage to any one display. Incoming clock pulses are applied to a diode and C2 is charged to the peak voltage of the incoming pulse, which, through the transistor base resistor, supplies sufficient drive to maintain the transistor on. For protection, ac coupling is used in case the clock fails in the High state. The transistor output is applied to the most significant bit of the scan decoder and in normal operation stays Low. If the clock fails, the transistor output goes High and the scan decoder addresses the two unused outputs. As shown, it operates satisfactorily from 1 kHz up with duty cycle pulse widths down to 10%. This circuit uses incandescent displays. Other displays may be substituted with appropriate drive circuitry.

8-DIGIT DISPLAY USING SHIFT REGISTERS AS A RECIRCULATING MEMORY


10k
0.1 µF
9601 O/S
ASTABLE MULTIVIBRATOR
SCAN COUNTER
9316 UP BINARY COUNTER
CEP CET CP PE P0 P1 P2 P3 TC MR Q0 Q1 Q2 Q3
9009
LOAD COMMAND
1/2 9024
SD CD J CP K Q
9009
A0 A1 A2 A3
9315 DECODER DR
0 1 2 3 4 5 6 7 8 9
SCAN DECODER
+60V
100k EACH
0 1 2 3 4 5 6 7
9300 SHIFT REGISTER
PE P0 P1 P2 P3 J CP K MR Q0 Q1 Q2 Q3
H3 G3 E3 F3 D3 C3 B3 A3
H2 G2 F2 E2 D2 C2 B2 A2
BCD INPUTS
H1 G1 F1 E1 D1 C1 B1 A1
H0 G0 F0 E0 D0 C0 B0 A0
MOST SIGNIFICANT DECADE = H0, H1, H2, H3
LEAST SIGNIFICANT DECADE = A0, A1, A2, A3
A0 A1 A2 A3 LT RBI
9307 7 SEGMENT DECODER
RBO a b c d e f g
9937 DTL HEX INVERTER
60V
10 k EACH
CONNECTED TO ALL OTHER DECADES AS SHOWN
LSB
8 SECONDARIES AT 1.6V PART NO. 4/2510
6 SECONDARIES AT 1.6V, 1 SECONDARY AT 70V PART NO. 4/2255A (TRANEX-MTN VIEW, CALIF.)
0 115V 1 2 3 4 5 6 7

Four rows of 8-bit, parallel loadable shift registers with outputs fed back to inputs form a recirculating memory, storing the four BCD bits in parallel and the eight characters serially. The four lines of serial information are fed into the 7-segment decoder circuitry which drives the display tubes. All tube anodes are in parallel. Multiplexing is achieved by sequentially switching the cathodes (filaments) to ground with the scan decoder. Eight BCD digits of parallel data are entered into the shift register when the Load input goes from High to Low. This transition is detected by the digital differentiator and activates the PE inputs for one clock period.

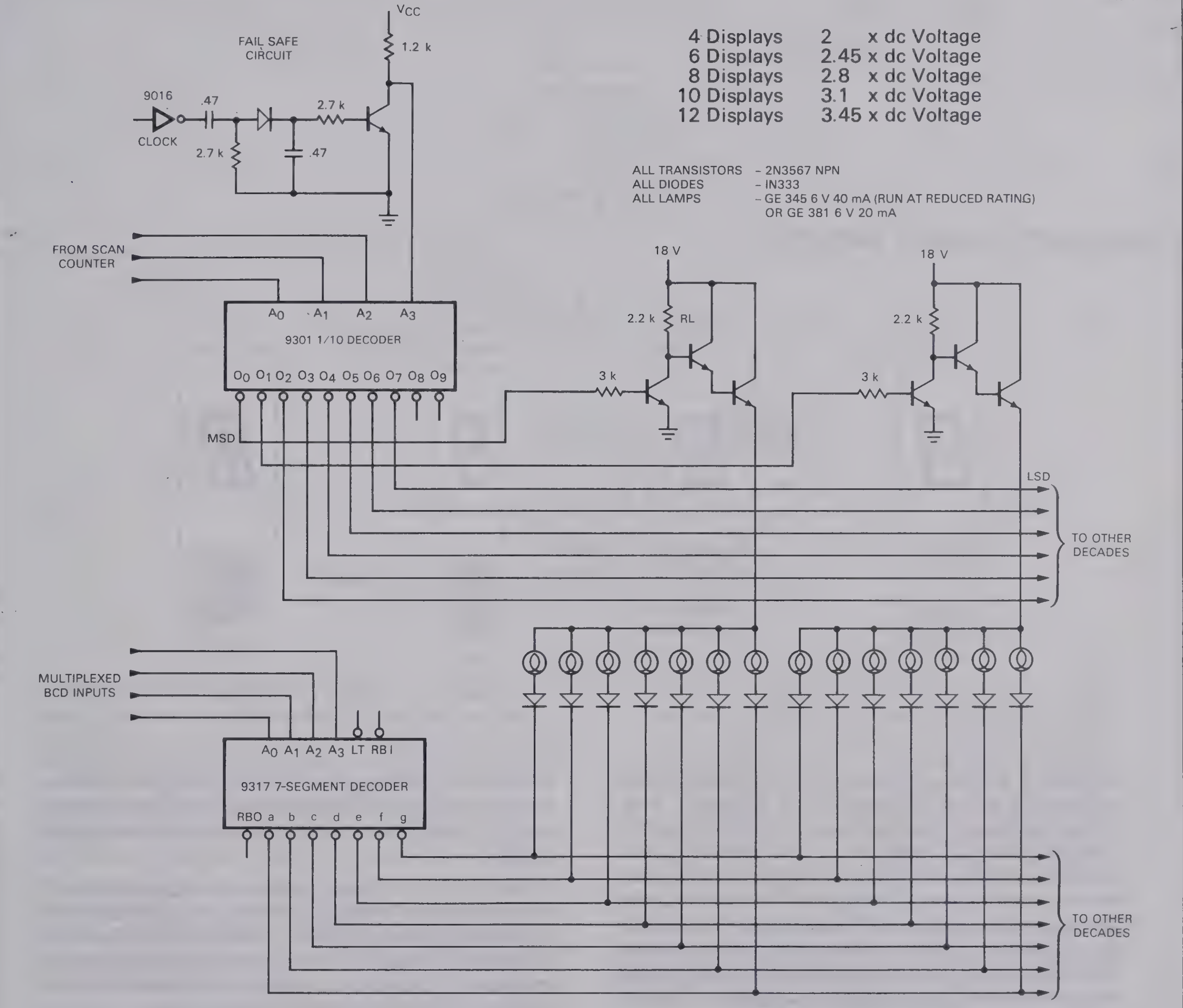
INCANDESCENT DIGITAL DISPLAY SYSTEMS

Seven-segment incandescent displays are the oldest of all digital displays and have been used extensively for many military applications. They are available with character heights from 1/4 inch to several feet in a variety of colors.

Since these displays employ incandescent lamps, high brightnesses can be achieved and they are available in a wide range of voltages and currents. The simplest 7-segment display is lamps placed behind seven slots on some opaque material.

Incandescent displays using seven aligned filaments to comprise a 7-segment character within one package are now available. One package is a small vacuum tube envelope; while another is a special, miniature flat package which inserts directly into a standard 16-lead Dual In-line socket. Many of the incandescent displays can be driven directly from a wide range of available decoder drivers (see selection guide).

MULTIPLEXING 7-SEGMENT INCANDESCENT DISPLAYS


VCC
1.2 k
FAIL SAFE CIRCUIT
9016
.47
2.7 k
CLOCK
2.7 k
.47
FROM SCAN COUNTER
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
MSD
18 V
2.2 k
RL
3 k
18 V
2.2 k
3 k
LSD
TO OTHER DECADES
MULTIPLEXED BCD INPUTS
A0 A1 A2 A3 LT RBI
9317 7-SEGMENT DECODER
RBO a b c d e f g
TO OTHER DECADES

4 Displays	2	x dc Voltage
6 Displays	2.45	x dc Voltage
8 Displays	2.8	x dc Voltage
10 Displays	3.1	x dc Voltage
12 Displays	3.45	x dc Voltage

ALL TRANSISTORS – 2N3567 NPN
ALL DIODES – IN333
ALL LAMPS – GE 345 6 V 40 mA (RUN AT REDUCED RATING) OR GE 381 6 V 20 mA

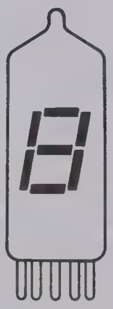
Multiplexed incandescent displays receive power in pulses for only a fraction of the complete scan period. For adequate brightness, voltage must be increased. The wattage for an incandescent lamp at a stated brightness will remain constant regardless of duty cycle or waveform shape, providing the multiplexing rate is faster then the time-constant of the filament. Power dissipation for a constant load is proportional to the square of the input voltage. Therefore, calculation of the required supply voltage for multiplexed operation is possible.

Multiplexed voltage $=\sqrt{\text{Number of displays}}$ x dc voltage of lamps, i.e., eight digits of multiplexed display readouts—6.3 volt lamps $\equiv \sqrt{8} \times 6.3 = 17.5$ volts.

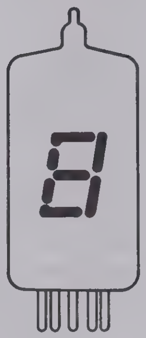
This table gives the factors for common numbers of multiplexed units. Voltage drops across the integrated circuits and switching transistors must be added to these figures. Some of the more advanced types use fiber optics or light-carrying beams to illuminate the seven segments.

FLUORESCENT DIGITAL DISPLAY SYSTEMS

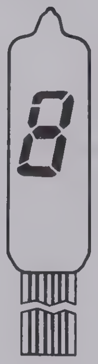
FLUORESCENT 7-SEGMENT DISPLAYS



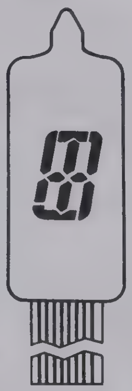
DIGIVAC
CONVENTIONAL
7 SEGMENT
FORMAT



SYLVANIA
FLUOROTRON



LEGI
DG10
NEC



LEGI
DG12
DG19
NEC
LD

Fluorescent displays are basically vacuum tube diodes or triodes in which the anodes form a visible 7 or 8-segment character. Each anode is coated with phosphor. A positive potential between anode and cathode (filament) accelerates electrons to the anode phosphor causing the anode to glow. The light emitted by the phosphors is a wide spectrum in the blue/green region which can be filtered to certain other colors with little light loss. The cathode (filament) is suspended in front of the display segments and is hardly visible since it is only heated sufficiently for electron emission.

Of the two types of fluorescent displays, triodes are somewhat easier to operate in a multiplexed mode because they have an extra, almost invisible control grid electrode between the filament and the anode. Multiple digit displays of up to ten digits in a single glass and metal enclosure are available using fluorescent triodes.

The main advantages of fluorescent displays are their economy, low power requirements, wide viewing angle, single plane readout, and minimum likelihood of erroneous readout with display failures. Two disadvantages are the need for a filament supply and their fragile structure.

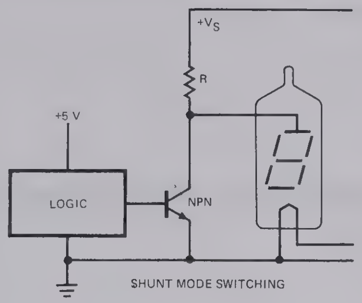
The selection of a decoder for fluorescent displays should take into account the fact that some displays have altered segment shapes as shown. Some decode formats provided by the decoders generate aesthetically unpleasant characters. Following are some points to note when selecting a fluorescent display decoder/driver.

9307	The decoder format is acceptable for all types of 7-segment displays. The decimal one appears on the right hand segments.
9358/9359	With these decoders, the decimal six looks poor on all except the Digivac* display. The decimal one is shown on the right hand side.

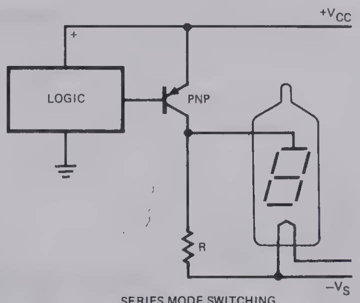
---

*Digivac is a registered trademark of Tung-Sol division of Wagner.

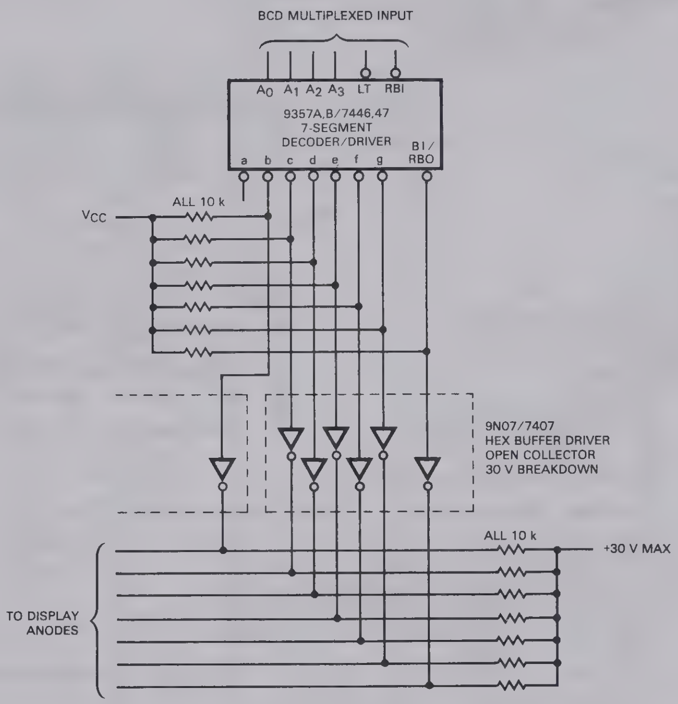
DRIVING FLUORESCENT DISPLAYS



SHUNT MODE SWITCHING



SERIES MODE SWITCHING



Driving fluorescent displays requires a 7-segment decoder and extra components. There are two basic methods of switching, the shunt method and the series method. As seen, the shunt method has a slightly better cost/part count when supply voltages of less than 30 V are involved because an open collector, hex inverter integrated circuit can be used. With supplies above 30 V, both systems require approximately the same number of parts; however, the series method is superior for power consumption. The shunt method shown draws maximum current when the display is blanked while the series method draws maximum current when all segments are illuminated.

The value of the load resistor selected for use with the shunt method is the result of a compromise between resistor power dissipation in the segments On and Off states. When the display is Off, the full supply voltage is across the resistor and maximum power is dissipated. While On, however, the only voltage drop is due to the segment current. Typical operating conditions for a non-multiplexed fluorescent display are 25 V and 0.5 mA to 1 mA per segment. The following example shows a driving method for 30 V supplies.

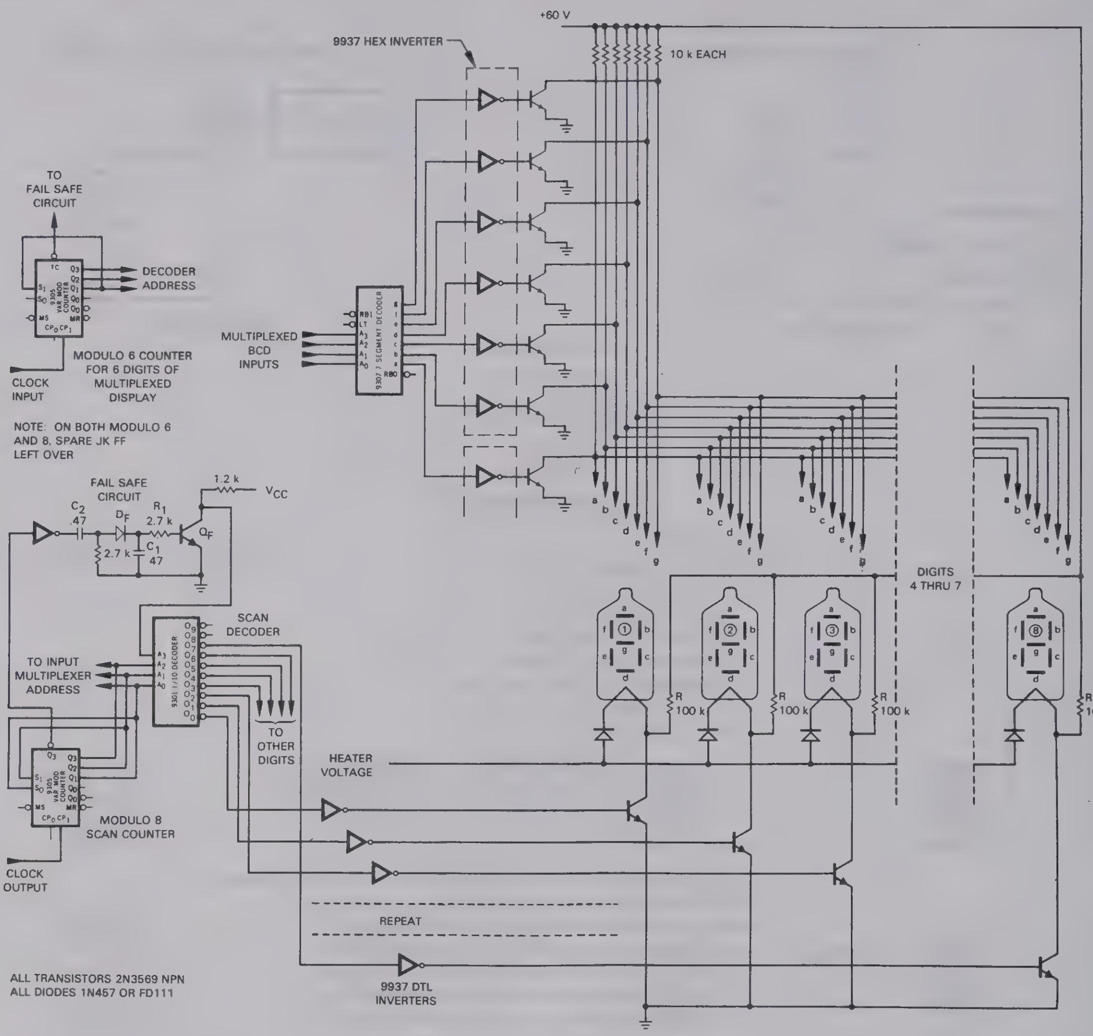
> With a 30 V supply, allowing for a 5 V drop across the resistors at 0.5 mA per segment, R = 10 kΩ. For displaying a decimal eight (all segments illuminated) total current = 7 segments x 0.5 mA = 3.5 mA. This means 105 mW supply power is required. In a blanked display, the current is 3 mA per segment (7) or 21 mA requiring 630 mW supply power. The Off to On power ratio is 630:105 or 6:1, worst case.

This would indicate that supply voltage regulation is required to maintain even brightness for all numerals displayed. A substantially better On to Off ratio is obtainable with a higher supply voltage.

> With a supply voltage of 60 V, allowing for a 35 V drop across R, 0.5 mA per segment = 70 kΩ. When displaying a decimal eight (3.5 mA), W = 60 x 3.5 or 210 mW. With the display blanked, current per segment is .85 mA or 6 mA. W = 60 x 6 or 360 mW. The Off to On power ratio is 360:210 or 1.7:1.

The higher supply voltage with larger resistance provides a more constant current source.

MULTIPLEXING FLUORESCENT DIODE DISPLAYS


+60 V
9937 HEX INVERTER
10 k EACH
TO
FAIL SAFE
CIRCUIT
DECODER
ADDRESS
9305 VAR MOD COUNTER
MODULO 6 COUNTER
FOR 6 DIGITS OF
MULTIPLEXED
DISPLAY
CLOCK
INPUT
NOTE: ON BOTH MODULO 6
AND 8, SPARE JK FF
LEFT OVER
MULTIPLEXED
BCD
INPUTS
9307 7 SEGMENT DECODER
FAIL SAFE
CIRCUIT
C_2
.47
D_F
R_1
2.7 k
2.7 k
C_1
47
Q_F
1.2 k
V_{CC}
SCAN
DECODER
9301 1/10 DECODER
TO INPUT
MULTIPLEXER
ADDRESS
TO
OTHER
DIGITS
HEATER
VOLTAGE
MODULO 8
SCAN COUNTER
CLOCK
OUTPUT
REPEAT
9937 DTL
INVERTERS
ALL TRANSISTORS 2N3569 NPN
ALL DIODES 1N457 OR FD111
DIGITS
4 THRU 7
R
100 k

The multiplexing system shown supplies the display heaters from convenient +5 V logic levels using a 1/8 (eight digit display) duty cycle. Heater power to the displays is supplied in pulse form and the thermal inertial of the filaments keeps them hot during the off period. Each of the display heaters has a transistor and a diode. The display is active when the transistor is conducting. When the transistor is Off, active resistor R pulls up the cathode to a +V potential, reverse biasing the diode leaving no potential across the display tube.

All the logic operates from a +5 V supply and thus it dictates a positive display voltage and a shunt method of switching. Because the display filaments are resistive, they have a given resistance value for a given emission value (ignoring switching losses) for either a pulsed or a dc power supply. The listed dc current and voltage specifications allow the calculation of the pulsed values.

$V_P = (\sqrt{N} \times V_F)$ + diode and switching losses

where: V_P = required voltage

N = number of parts in scan cycle

V_F = dc filament voltages of the display

Current is determined by substituting

I_F filament current

For the 8-digit example shown, the required pulsed voltage calculation is given.

$$V_P = (\sqrt{8} \times 1.4\text{ V}) + .9\text{ V} = 4.9\text{ V}$$

where: 8 = number of digits,

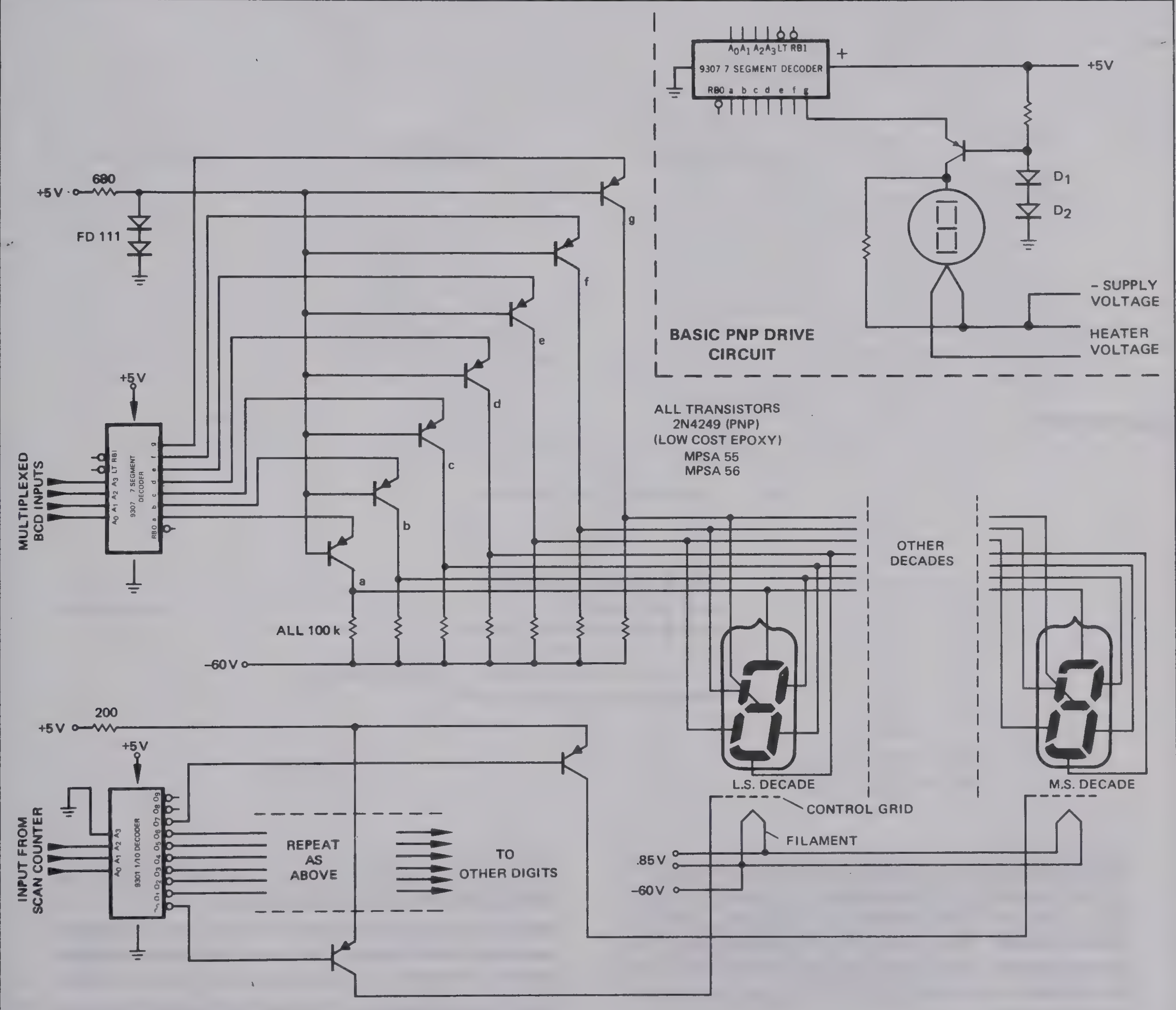
1.4 = filament voltage,

.9 = diode/switching loss (.7 V/diode and .2 V/transistor)

This circuit is easily changed to a 6-digit system by changing the scan counter to a Modulo 6.

A failsafe circuit is added to the most significant digit of the scan decoder to prevent full filament supply voltage from being applied to any one heater should the clock fail. With the counter operating, pulses are supplied to diode D_F and the R-C network, biasing transistor Q_F on. Failure removes the bias on Q_F and the input code to the scan decoder addresses unused outputs 8 and 9. This circuit operates with clock frequencies up to 20 kHz.

MULTIPLEXING FLUORESCENT TRIODE DISPLAYS


A0 A1 A2 A3 LT RBI
9307 7 SEGMENT DECODER
RBO a b c d e f g
+
+5V
D1
D2
– SUPPLY VOLTAGE
HEATER VOLTAGE
BASIC PNP DRIVE CIRCUIT
+5V
680
FD 111
g
f
e
d
c
b
a
+5V
MULTIPLEXED BCD INPUTS
A0 A1 A2 A3 LT RBI
9307 7 SEGMENT DECODER
RBO a b c d e f g
ALL 100 k
–60 V
OTHER DECADES
L.S. DECADE
M.S. DECADE
CONTROL GRID
FILAMENT
.85 V
–60 V
+5V
200
+5V
INPUT FROM SCAN COUNTER
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
REPEAT AS ABOVE
TO OTHER DIGITS

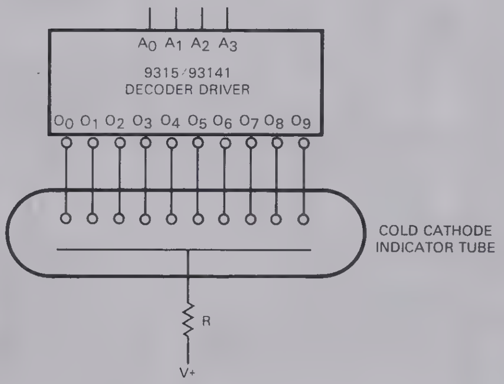
ALL TRANSISTORS
2N4249 (PNP)
(LOW COST EPOXY)
MPSA 55
MPSA 56

The only decoder/driver which provides the correct decode format for fluorescent triode displays is the 9307 and it requires the use of extra drive components. The most economical method uses pnp drive transistors with negative supply voltage. The active high output of the 9307 biases the transistor emitter more positive than the base, turning the transistor on. Diodes D1 and D2 insure that the transistor is back biased in the Off position (right). Collector resistors are used so anode capacitance charges can be leaked off quickly.

Control grids of fluorescent displays are activated to the same potential as the anodes to obtain full brightness. They are activated by a pnp switch transistor. This is driven directly by the output of the 9301 scan decoder which has active low outputs. The base-emitter current on the pnp transistor is limited by a single resistor in common with all the pnp emitters.

GAS DISCHARGE DIGITAL DISPLAY SYSTEMS

DESCRIPTION



Gas discharge displays, otherwise known as cold cathode, are available in a wide variety of shapes and sizes. The oldest and most familiar is the NIXIE*, with the newest sample being the Burroughs Panaplex or the Sperry displays. All of these displays require a high supply voltage in the range of 80 to 200 V to cause ionization for display.

The nixie-type tube is not a 7-segment display but a 1-of-10 multiplane display with ten stacked cathodes shaped as characters mounted behind an almost invisible anode in a glass envelope filled with neon gas. A high potential between the anode and one of the cathode characters causes an ionization to occur and the character glows a dull red. These displays are available in a wide variety of shapes and sizes from miniature sizes with character heights of just over 1/4 inch to jumbo sizes with 2 inch characters.

The advantages of this type of display are: availability, reliability, wide selection of sizes and styles, and low power consumption. Special characters are available.

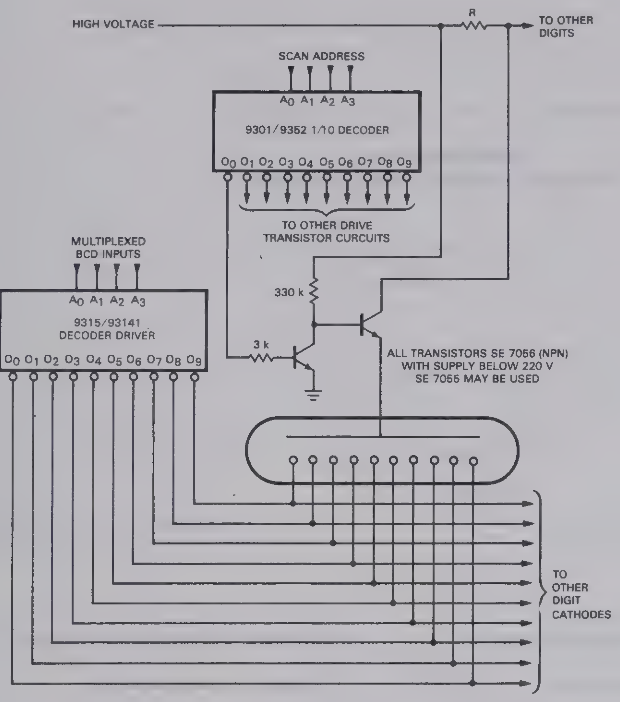
Some disadvantages of this display system are: high voltage requirements, multiplexing difficulty, multiplane display readout, and limited color selection.

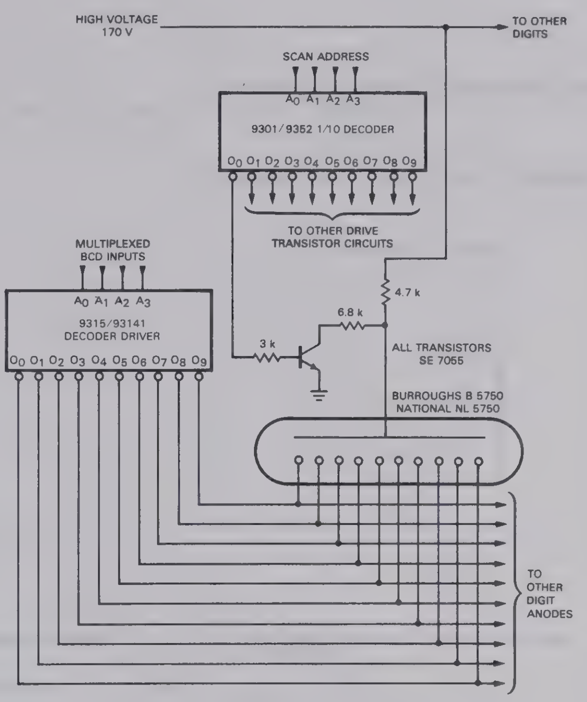
Cold cathode displays require a 1-of-10 decoder/driver with high voltage breakdown npn transistors on the outputs. The transistors must have low leakage characteristics in the Off mode since leakage causes a background glow. Voltage across the displays after ionization is controlled by a series resistor, as shown above. This resistance value is given by the display manufacturers.

---

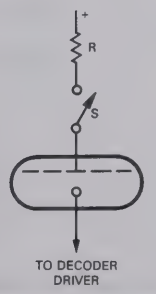
*NIXIE is a registered trademark of Burroughs Corporation

MULTIPLEXING GAS DISCHARGE DISPLAYS


HIGH VOLTAGE
R
TO OTHER DIGITS
SCAN ADDRESS
A0 A1 A2 A3
9301/9352 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
TO OTHER DRIVE TRANSISTOR CURCUITS
MULTIPLEXED BCD INPUTS
A0 A1 A2 A3
9315/93141 DECODER DRIVER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
330 k
3 k
ALL TRANSISTORS SE 7056 (NPN) WITH SUPPLY BELOW 220 V SE 7055 MAY BE USED
TO OTHER DIGIT CATHODES

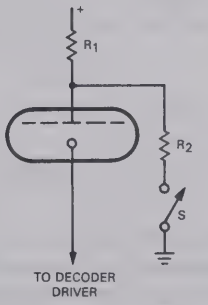

HIGH VOLTAGE 170 V
TO OTHER DIGITS
SCAN ADDRESS
A0 A1 A2 A3
9301/9352 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
TO OTHER DRIVE TRANSISTOR CIRCUITS
MULTIPLEXED BCD INPUTS
A0 A1 A2 A3
9315/93141 DECODER DRIVER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
4.7 k
6.8 k
3 k
ALL TRANSISTORS SE 7055
BURROUGHS B 5750 NATIONAL NL 5750
TO OTHER DIGIT ANODES

SERIES METHOD


+
R
S
TO DECODER DRIVER

NOTE: VALUE R FROM MANUFACTURERS SPECIFICATIONS.

SHUNT METHOD


+
R1
R2
S
TO DECODER DRIVER

NOTE: RATIO OF R_1 AND R_2 IS SUCH THAT WHEN THE SWITCH IS CLOSED, ANODE POTENTIAL IS LOWER THAN IONIZATION POTENTIAL.

It is necessary to sequentially switch the high voltage anode supply when multiplexing this type of display so that only one digit is addressed at a time. There are two basic approaches for accomplishing this, the series method and the shunt method. Part count and economy are not considerations since both methods use identical hardware. The series method offers an advantage with lower power dissipation but places higher voltage and leakage requirements on the transistors. Leakage is not as important in the shunt method.

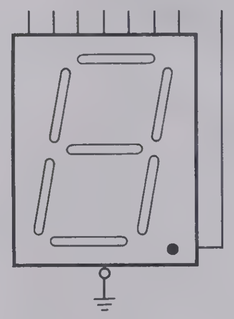
There are several different 7-segment gas discharge displays available, ranging widely in configuration and complexity. The oldest type are simply seven neon lamps arranged behind seven slots in an opaque material. Another form of 7-segment gas discharge displays is a small tube envelope which encloses a specially constructed 7-segment unit with a character height of 7/16 of an inch. The newer types are Burroughs Panaplex and the Sperry units. Burroughs Panaplex is a multi-digit unit and, therefore, must be multiplexed. Special drive considerations are required, best obtained from the manufacturer's literature. Sperry displays can be either driven directly or can be multiplexed if desired.

Neon 7-segment displays require a resistor in series with each of the cathodes. A separate resistor is required because the ionization start-up voltage required is higher than that needed to maintain ionization. Once ionization takes place the resistor lowers the voltage applied between segment cathode and anode thus maintaining correct segment current. Neon 7-segment displays can be multiplexed with the same techniques used for 1-of-10 cold cathode displays previously described.

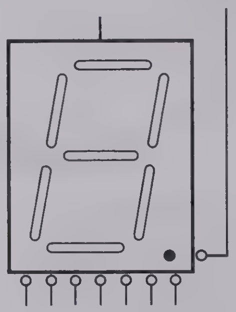
LED DIGITAL DISPLAY SYSTEMS

LIGHT EMITTING DIODE DISPLAYS

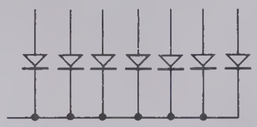
COMMON CATHODE
(DECODER/DRIVER SOURCES CURRENT)



COMMON ANODE
(DECODER/DRIVER SINKS CURRENT)

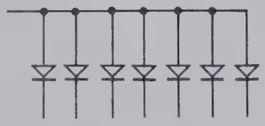


SINGLE DIODE LED DISPLAY CIRCUIT



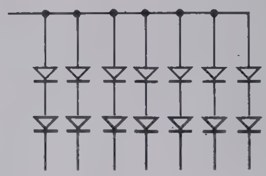
	GaP	GaAsP
1 DIODE	≃2.1 V	≃1.65 V

SINGLE DIODE LED DISPLAY CIRCUIT



	GaP	GaAsP
1 DIODE	≃2.1 V	≃1.54 V

DUAL DIODE LED DISPLAY CIRCUIT



	GaP	GaAsP
2 DIODES	≃4.2 V	≃ 3.4 V

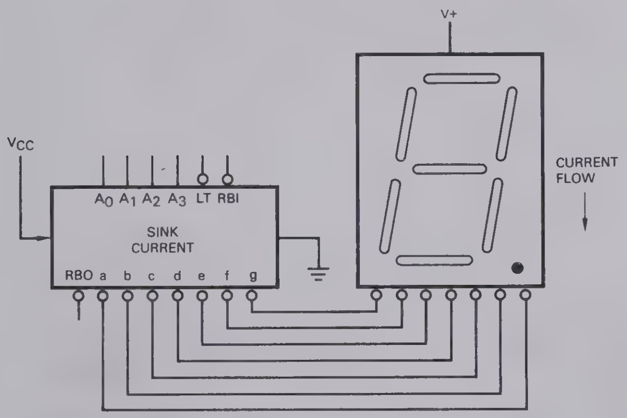
Light emitting diodes (LEDs) are now found in a wide variety of portable electronic equipment including pocket calculators, electronic stop watches, thermometers, digital voltmeters, and similar equipment. They have gained popularity by virtue of their compact size, voltage and current compatibility with integrated circuit technology, and solid-state reliability. Typically small in character size, they can be made up to 0.8 inch using hybrid construction techniques.

The standard LED display color is a bright, penetrating monochromatic red with sharply defined characters which appear about twice as large as their actual size. Green and yellow displays are also offered, but are somewhat more expensive. The primary disadvantage inherent with LEDs is a narrow spectrum of light output. This makes them not only difficult to filter for high ambient light applications, but impossible to filter for alternate colors. A polarized filter improves viewing under normal light circumstances. See the Selection Guide for a list of currently available LED displays.

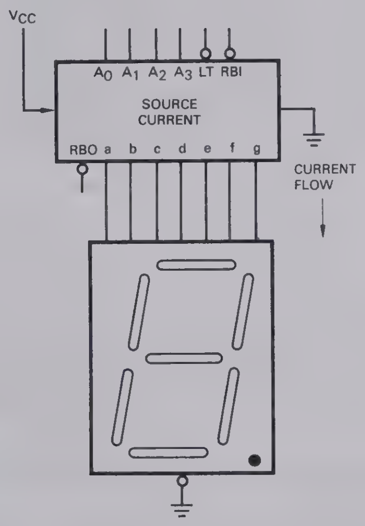
LED displays use either gallium phosphide (GaP) or gallium arsenide phosphide (GaAsP). Both types seem to have equal advantages and disadvantages. They are available in two character formats, a 5 x 7 dot matrix which can display alphanumerics and lower cost, 7-segment displays. LED 7-segment displays use either common anode or common cathode techniques. The 0.1 inch character height calls for the common cathode. Hybrid construction can be either, but the common anode has more popularity. With the common cathode method, the decoder/driver must be able to source current into the display. The common anode variety requires the decoder/driver to sink current, which is presently easier.

Electrically, the LED is the same as the familiar solid-state diode with its inherent characteristics although with a higher forward voltage drop. Below the knee of the I/V curve, they pass little current. Above the knee, current increases linearly with voltage. Thus, uniform brightness of each segment and digit requires a constant current source rather than constant voltage. Light output increases linearly with current for low values until light saturation is reached and further current does not increase light output. Current should be limited to avoid device failure.

DRIVING LED DISPLAYS



COMMON ANODE



COMMON CATHODE

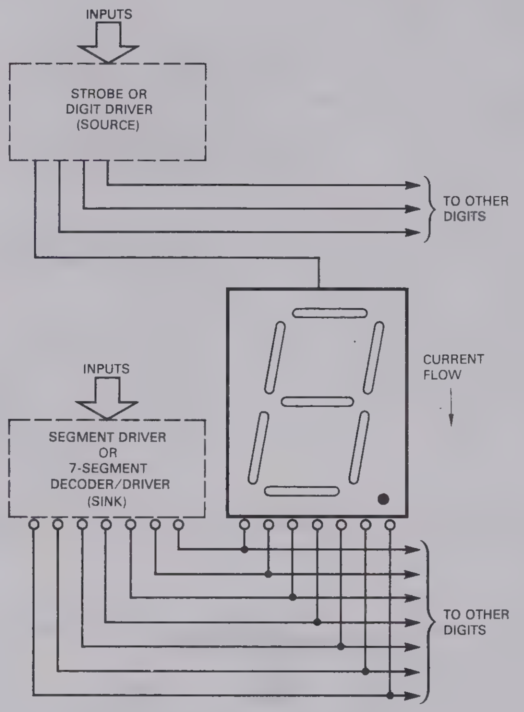
LED displays have electrical and light output rise times in the nanosecond region. This means they can be operated at a low duty cycle with high scanning rates when multiplexed. Scanning rates above 1 kHz are recommended. As many as 24 or more displays can be multiplexed. Because the human eye tends to peak-detect, a low duty cycle with high peak current pulsed into LED displays provides high apparent brightness.

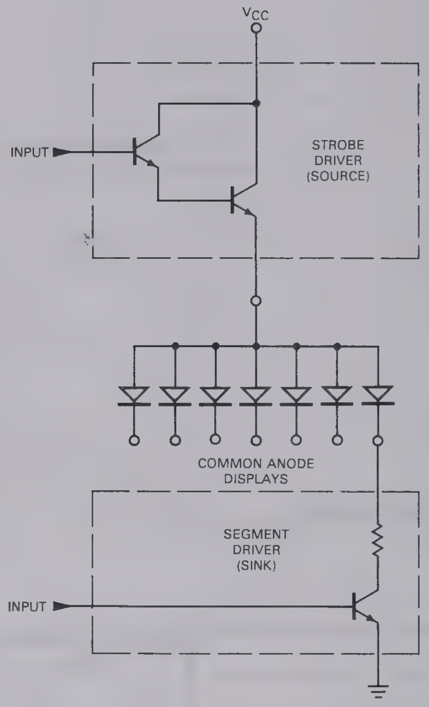
Drive requirements for 7-segment decoder/drivers are not as stringent as those for character or digit address where the total pulsed current of all seven segments must pass through. Ideally, the decoder/driver is a constant current source or sink, depending upon the device. The digit address is a low resistance switch. Resistors may be required for current control to the displays, depending on displays and decoders used.

In some applications, a common cathode LED display, when driven by the fixed current output of a 9368 decoder/driver/latch, is too bright. Since the 9368 has no provision for output current adjustment, an intensity modulation technique is used to reduce both brightness and power dissipation — and thus, heat. Because the human eye tends to peak detect, and because decreasing current to the display would decrease its efficiency, a pulsed duty cycle technique is superior to a "brute force" current reduction. A thirty percent savings in power can be realized without a significant loss of brightness. When a reduction in brightness is desired, further savings of power is possible.

LED DIGITAL DISPLAY SYSTEMS

MULTIPLEXING COMMON ANODE LED DISPLAYS


INPUTS
STROBE OR DIGIT DRIVER (SOURCE)
TO OTHER DIGITS
CURRENT FLOW
INPUTS
SEGMENT DRIVER OR 7-SEGMENT DECODER/DRIVER (SINK)
TO OTHER DIGITS


V_{CC}
INPUT
STROBE DRIVER (SOURCE)
COMMON ANODE DISPLAYS
SEGMENT DRIVER (SINK)
INPUT

FND 507	Fairchild Optoelectronics
7730 Series	Hewlett Packard
DL 1A, 6, 8, 10A, 62, 707	Litronix
MAN 1, 6A, 10, 1001	Monsanto
TIL 302	Texas Instrument
SLA 1, 3, 4, 7	Opcoa

7-SEGMENT DECODER/DRIVERS (SINK CURRENT)

9317 7-segment decoder/driver sinks 40 mA and requires seven current limiting resistors. TTL inputs.

9357 7-segment decoder/driver sinks 40 mA and requires seven current limiting resistors. TTL inputs.

9370 7-segment decoder/driver/latch sinks 40 mA and requires seven current limiting resistors. TTL inputs.

SEGMENT DRIVERS (SINK CURRENT)

9662 Quad LED digit/lamp driver sinks 600 mA has Inverting Enable input control for inverting or non-inverting operation. If used as a segment driver, two 9662 devices and seven current limiting resistors are needed to drive 7-segment displays. MOS/TTL compatible inputs.

9663 Hex LED digit/lamp driver sinks 600 mA is the same as 9662 but with six outputs.

STROBE OR DIGIT DRIVERS (SOURCE CURRENT)

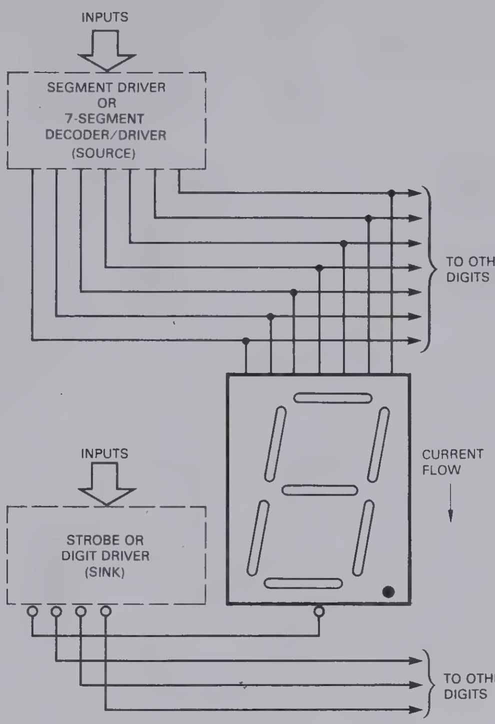
9661 Programmable current quad segment driver sources 5-75 mA. MOS/TTL compatible inputs.

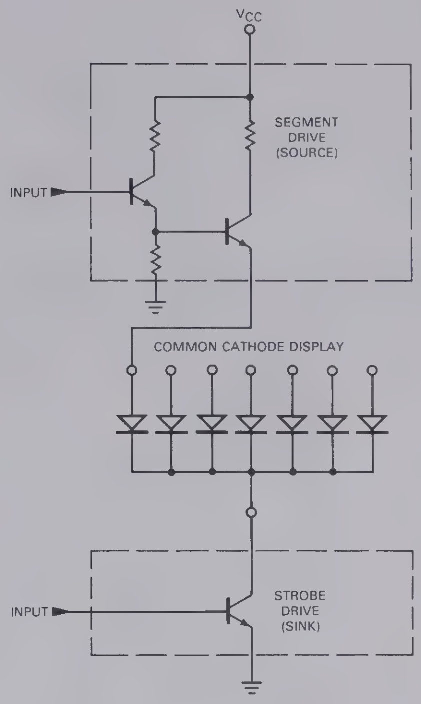
75491 Quad segment driver sinks or sources 50 mA. Each section has two transistors, Darlington connected, with uncommitted emitter and collector.

Transistors, High current npn or pnp transistors with resistors can be used for strobe or digit drive.

LED DIGITAL DISPLAY SYSTEMS

MULTIPLEXING COMMON CATHODE LED DISPLAYS


INPUTS
SEGMENT DRIVER OR 7-SEGMENT DECODER/DRIVER (SOURCE)
TO OTHER DIGITS
CURRENT FLOW
INPUTS
STROBE OR DIGIT DRIVER (SINK)
TO OTHER DIGITS



FND 70, 500	Fairchild Optoelectronics
FNA 30, 45	Fairchild Optoelectronics
DA 1007, DA 1009	Antex
R7M	Bowmar
4700 Series	Hewlett Packard
DL 4, 33, 34, 704	Litronix
NSN 33, 133	National Semiconductor
MAN 3, 4	Monsanto

7-SEGMENT DECODER/DRIVERS (SOURCE CURRENT)

9368 7-segment decoder/driver/latch sources 19 mA @ 1.7 V output. Hexadecimal decode format. TTL inputs.

9369 7-segment decoder/driver/latch sources 50 mA @ 3.4 V output and requires current limiting resistors. TTL inputs.

9660 7-segment decoder/driver with decimal point drive, sources 5-50 mA. Output current is controlled by one external resistor. MOS/TTL compatible inputs.

SEGMENT DRIVERS (SOURCE CURRENT)

9661 Programmable current quad segment driver (two per system required) sources 5-75 mA @ 3 V output. One output can be reduced by approximately 1/2 for decimal point drive. One external resistor determines the output currents of both devices.

75491 Quad segment driver sinks or sources 50 mA. Inverting when used in sink mode. Two devices and seven current limiting resistors are required to drive seven segments. Each section has two transistors, Darlington connected, with uncommitted emitter and collector.

75492 Hex LED/lamp driver sinks 250 mA. Each section has two transistors, Darlington connected, with uncommitted collector.

STROBE OR DIGIT DRIVERS (SINK CURRENT)

9662 Quad LED digit/lamp driver sinks 600 mA, has Inverting Enable input control for inverting or non-inverting operation. MOS/TTL compatible inputs.

9663 Hex LED digit/lamp driver sinks 600 mA is the same as 9662 with six outputs.

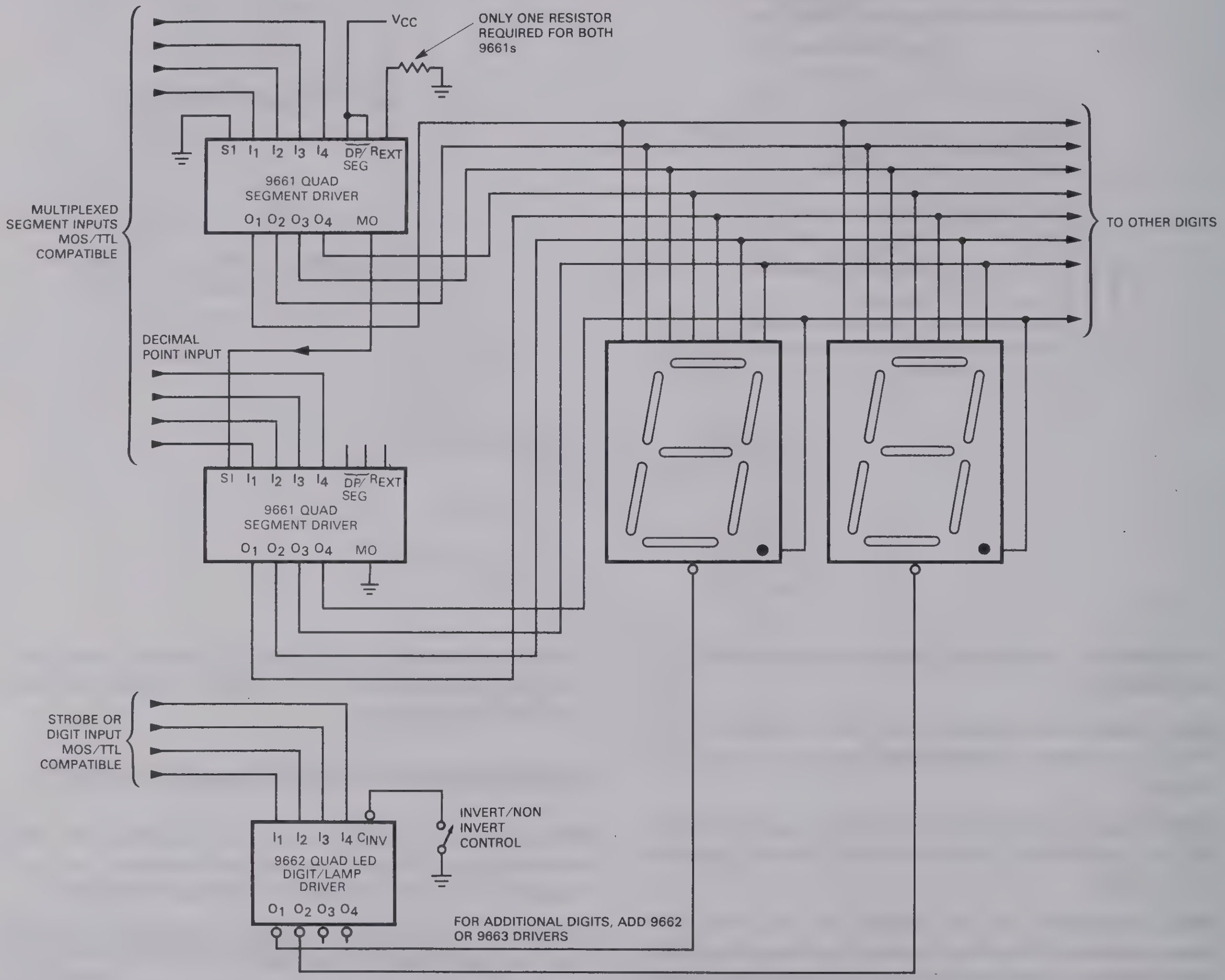
9345, 7445 1-of-10 decoder/driver sinks 80 mA. TTL inputs.

93145, 74145 1-of-10 decoder/driver sinks 80 mA. TTL inputs.

75491 Quad segment driver sinks or sources 50 mA. Each section has two transistors, Darlington connected, with uncommitted emitter and collector.

75492 Hex LED/lamp driver sinks 250 mA. Each section has a Darlington transistor with uncommitted collector.

MULTIPLEXED COMMON CATHODE LED CALCULATOR DISPLAY


VCC
ONLY ONE RESISTOR REQUIRED FOR BOTH 9661s
S1 I1 I2 I3 I4 DP/SEG REXT
9661 QUAD SEGMENT DRIVER
O1 O2 O3 O4 MO
MULTIPLEXED SEGMENT INPUTS MOS/TTL COMPATIBLE
TO OTHER DIGITS
DECIMAL POINT INPUT
STROBE OR DIGIT INPUT MOS/TTL COMPATIBLE
INVERT/NON INVERT CONTROL
I1 I2 I3 I4 CINV
9662 QUAD LED DIGIT/LAMP DRIVER
FOR ADDITIONAL DIGITS, ADD 9662 OR 9663 DRIVERS

The multiplexing circuit shown uses two 9661 quad segment drivers. The four outputs of the first unit drive segments a, b, c, and d of each display. The four outputs of the second 9661 drive segments e, f, g, and the decimal point of each display.

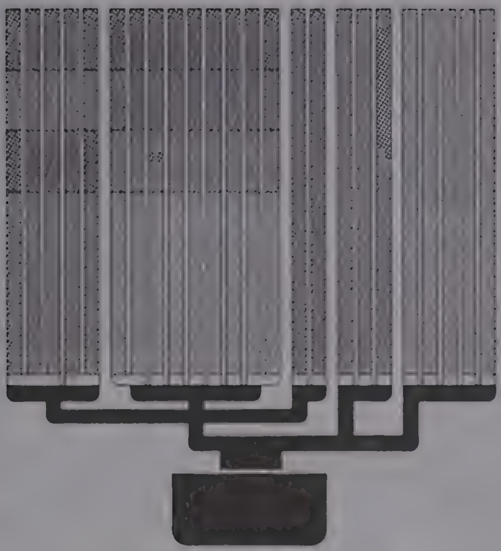
In this circuit, the two 9661 segment drivers can be replaced with one 9660 7-segment decoder/driver when current required is less than 50 mA @ 3 V and BCD to 7-segment decoding is required.

Other segment/digit/lamp drivers can be used depending on display requirements.

9661 — Programmable current quad segment driver can source current from 5 to 75 mA @ 3 V. Output current is determined by one external resistor R_{EXT} to ground. If the DP/Segment inputs are not connected to V_{CC}, the current at O_4 is reduced by 50%, which is suitable for driving the decimal point. All inputs are TTL/MOS compatible.

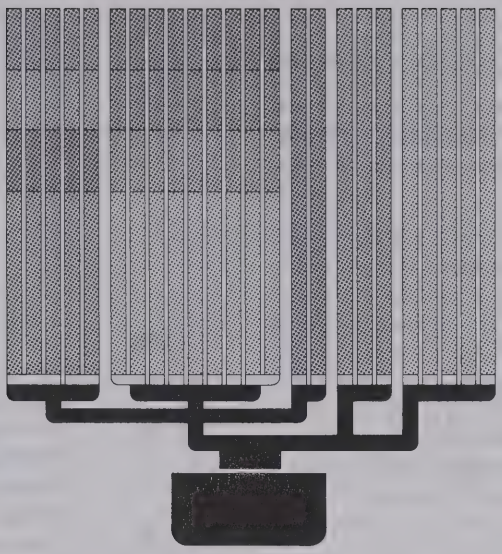
9662 — Quad LED digit/lamp driver is capable of sinking 600 mA on each of four outputs and all inputs are TTL/MOS compatible. Inverting operation is selected by grounding the Control Invert input.

9663 — Hex LED digit/lamp driver is the same as the 9662 but with six outputs.



Encoders

Serial and Parallel Expansion
Switch Encoder
Linear Priority Encoder
Code Conversion
Computer Priority Interrupts
Simultaneous A/D Conversion
D/A Conversion, Binary Rate Multiplier
D/A Conversion, BCD Rate Multiplier
Single Zero Detection



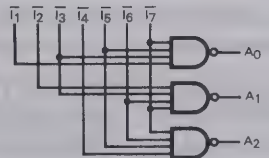
ENCODERS

INTRODUCTION

Encoders are circuits with many inputs that generate the address of the active input. If a system design guarantees only one encoder input active, the encoder logic is very simple and can be implemented with gates.

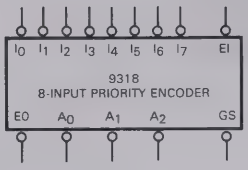
If several inputs can be active at one time, a simple encoder would generate the logic OR of their addresses, which is probably undesirable (i.e., inputs 2 and 4 active would generate address 6).

A priority encoder generates the address of the active input with the highest priority. The priority is pre-assigned according to the position at the inputs. This chapter describes the 9318 8-input priority encoder and applications.


$\overline{I}_1$
$\overline{I}_2$
$\overline{I}_3$
$\overline{I}_4$
$\overline{I}_5$
$\overline{I}_6$
$\overline{I}_7$
A_0
A_1
A_2

9318 8-INPUT PRIORITY ENCODER

DESCRIPTION



LEADS		LOADING
$\overline{I}_0$	Priority (Active Low) Input	1 UL
$\overline{I}_1 - \overline{I}_7$	Priority (Active Low) Inputs	2 UL
$\overline{EI}$	Enable (Active Low) Input	2 UL
$\overline{EO}$	Enable (Active Low) Output	5 UL
$\overline{GS}$	Group Signal (Active Low) Output	6 UL
$\overline{A}_0, \overline{A}_1, \overline{A}_2$	Address (Active Low) Outputs	10 UL

	$\overline{EI}$	$\overline{I}_0$	$\overline{I}_1$	$\overline{I}_2$	$\overline{I}_3$	$\overline{I}_4$	$\overline{I}_5$	$\overline{I}_6$	$\overline{I}_7$	$\overline{GS}$	$\overline{A}_0$	$\overline{A}_1$	$\overline{A}_2$	$\overline{EO}$
(DISABLED)	H	X	X	X	X	X	X	X	X	H	H	H	H	H
(NO ACTIVE INPUT)	L	H	H	H	H	H	H	H	H	H	H	H	H	L
	L	X	X	X	X	X	X	X	L	L	L	L	L	H
	L	X	X	X	X	X	X	L	H	L	H	L	L	H
	L	X	X	X	X	X	L	H	H	L	L	H	L	H
	L	X	X	X	X	L	H	H	H	L	H	H	L	H
	L	X	X	X	L	H	H	H	H	L	L	L	H	H
	L	X	X	L	H	H	H	H	H	L	H	L	H	H
	L	X	L	H	H	H	H	H	H	L	L	H	H	H
	L	L	H	H	H	H	H	H	H	L	H	H	H	H

The 9318 8-input priority encoder is a multipurpose device useful in a wide variety of applications such as priority encoding, priority control, decimal or binary encoding, code conversion, analog-to-digital and digital-to-analog conversion. A priority encoder can improve computer systems by providing the computer with high speed hardware priority interrupt capabilities.

The 9318 provides three bits of binary coded output representing the position of the highest order input, along with an output indicating the presence of any input. It is easily expanded through input and output enables to provide priority encoding over many bits.

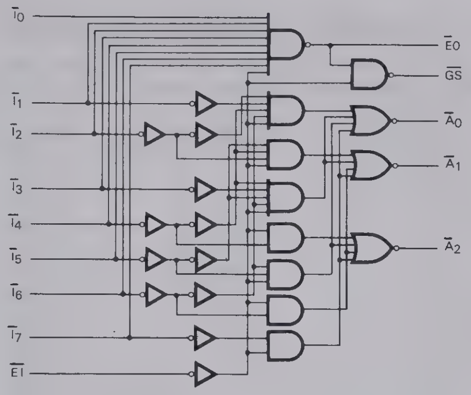
The 9318 is a member of the 9300 TTL/MSI family and is compatible logically and electrically with other members of the family. It features active pull up outputs which provide high speed and excellent noise margins, and has input clamp diodes to reduce negative line transients.

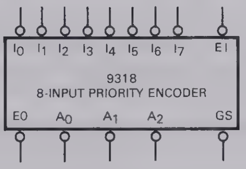
The logic symbol, lead functions, loading rules, and truth table for the 9318 are shown above. The 9318 accepts eight active Low inputs ($I_0 - I_7$) and produces a binary weighted output code ($\overline{A}_0\overline{A}_1\overline{A}_2$) representing the position of the highest order active input. A priority is thus assigned to each input ($\overline{I}_7$ has the highest priority). Thus, when two or more inputs are simultaneously active, the input with the highest priority is encoded and the other inputs are ignored. In addition, all inputs are OR tied to provide a group signal indicating the presence of any Low input signal. This group signal is Low whenever any input is Low and the encoder is enabled.

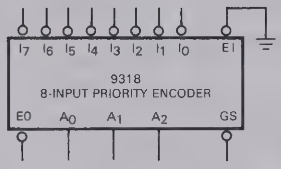
The drive requirements for all priority inputs and the enable input are two or less TTL loads, thereby reducing the output needed from the driving circuitry. The address outputs can drive 10 unit loads, and the group signal and enable outputs drive six and five unit loads respectively.

9318 8-INPUT PRIORITY ENCODER

OPERATION


$\overline{I_0}$
$\overline{I_1}$
$\overline{I_2}$
$\overline{I_3}$
$\overline{I_4}$
$\overline{I_5}$
$\overline{I_6}$
$\overline{I_7}$
$\overline{EI}$
$\overline{EO}$
$\overline{GS}$
$\overline{A_0}$
$\overline{A_1}$
$\overline{A_2}$


I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS


I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS

When High, the Enable input ($\overline{EI}$) overrides all inputs and forces all outputs High. The active Low Enable output indicates that the encoder is enabled and no active signal has been applied. The combination of Enable input and Enable output permits series expansion of priority encoding to many levels with little additional logic, explained later. The truth lable lists all the input combinations and the resulting outputs. All inputs and outputs are active Low, providing logic compatibility with other Fairchild TTL devices.

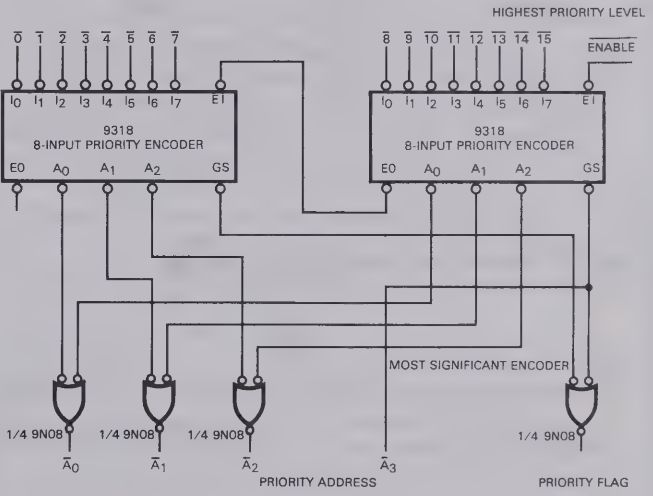
The 9318 priority encoder also can be represented with active High address outputs, but in this case disabling the input generates an address 7. Also, input zero has the highest priority. Because the 9318 is a combinational network, a wrong address can appear during input transients. This must be considered when strobing the outputs. Moreover, when all priority inputs are High, a High-to-Low Enable change can cause a transient on the Group Signal output.

PRIORITY ENCODER EXPANSION

The 9318 priority encoder can be expanded in series or in parallel. Expansion in series requires a minimum of components while expansion in parallel offers the highest operaing speed.

The expanded priority encoder generates the binary address of the highest order priority input in two or more levels. The inputs are applied to what can be considered the first level of priority encoders. Since a binary code is cyclic, the output of each encoder potentially represents the three least significant bits of the output. Series and parallel expansion differ in the way the proper encoder and its output code are selected. In both cases, the group signals of the encoders go through a second level of encoding to determine the more significant group. Two levels of encoding are required for up to 64 inputs, and three levels are required for 65 to 512 inputs.

SERIES EXPANDED ENCODERS

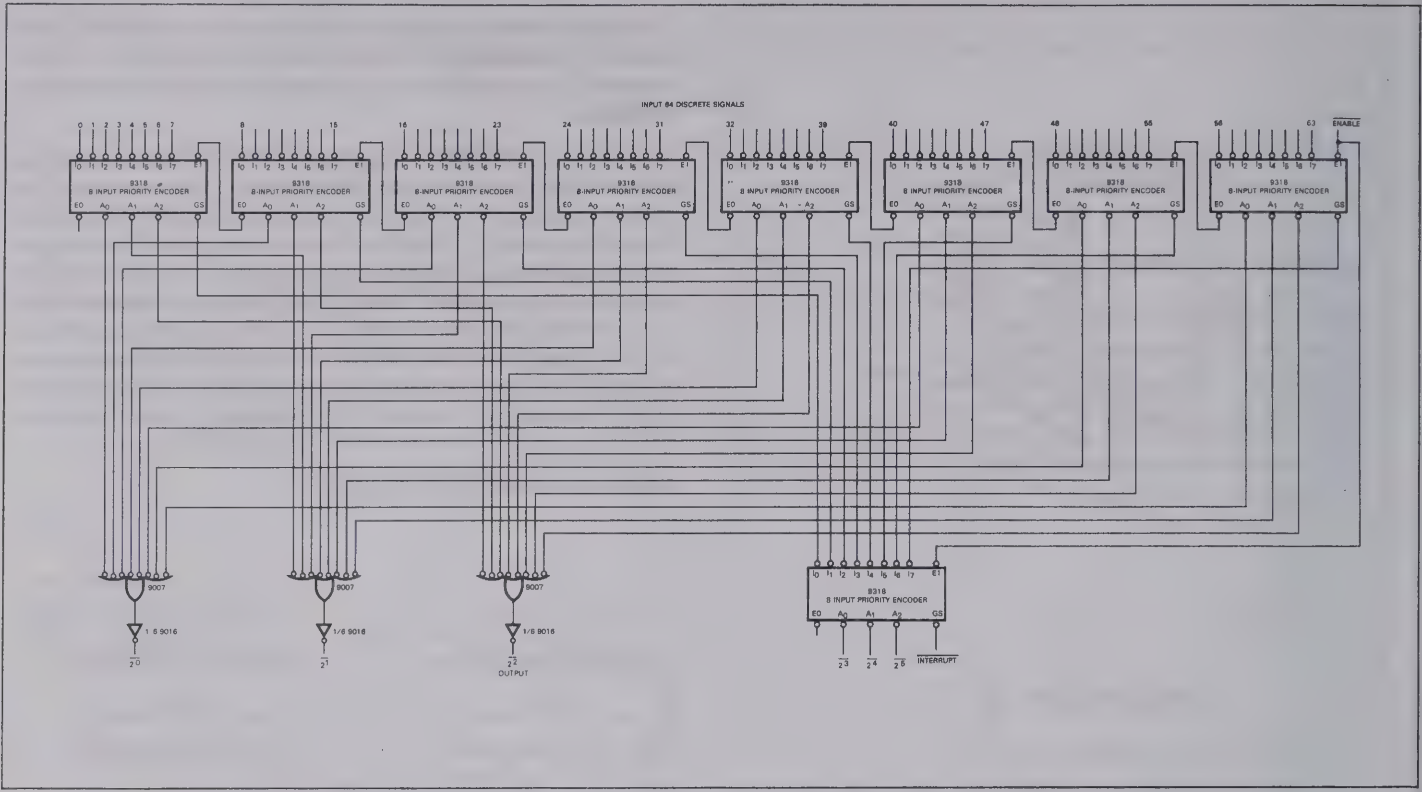

HIGHEST PRIORITY LEVEL
$\overline{0}$ $\overline{1}$ $\overline{2}$ $\overline{3}$ $\overline{4}$ $\overline{5}$ $\overline{6}$ $\overline{7}$
$\overline{8}$ $\overline{9}$ $\overline{10}$ $\overline{11}$ $\overline{12}$ $\overline{13}$ $\overline{14}$ $\overline{15}$
$\overline{ENABLE}$
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
MOST SIGNIFICANT ENCODER
1/4 9N08
1/4 9N08
1/4 9N08
1/4 9N08
$\overline{A_0}$
$\overline{A_1}$
$\overline{A_2}$
$\overline{A_3}$
PRIORITY ADDRESS
PRIORITY FLAG

In this 16-input expanded encoder, and in the 64-input version shown next, the priority encoders are enabled in series; the Enable output of the more significant encoder is connected to the Enable input of the next less significant encoder. This allows the highest order encoder with an active Low input to disable all less significant encoders. The selection of the three least significant bits is simple since this particular encoder is the only one with active outputs; all less significant encoders are disabled and all more significant encoders have no active inputs and consequently no active outputs. Therefore the appropriate address outputs can be ORed with active Low input and output OR gates (ANDs) to generate the least significant bits. The group signals of each first level encoder are encoded, if necessary, to provide the most significant bits of the output. In a series array, only the highest order encoder with an active input has a Group Signal output and its address can be generated with gates.

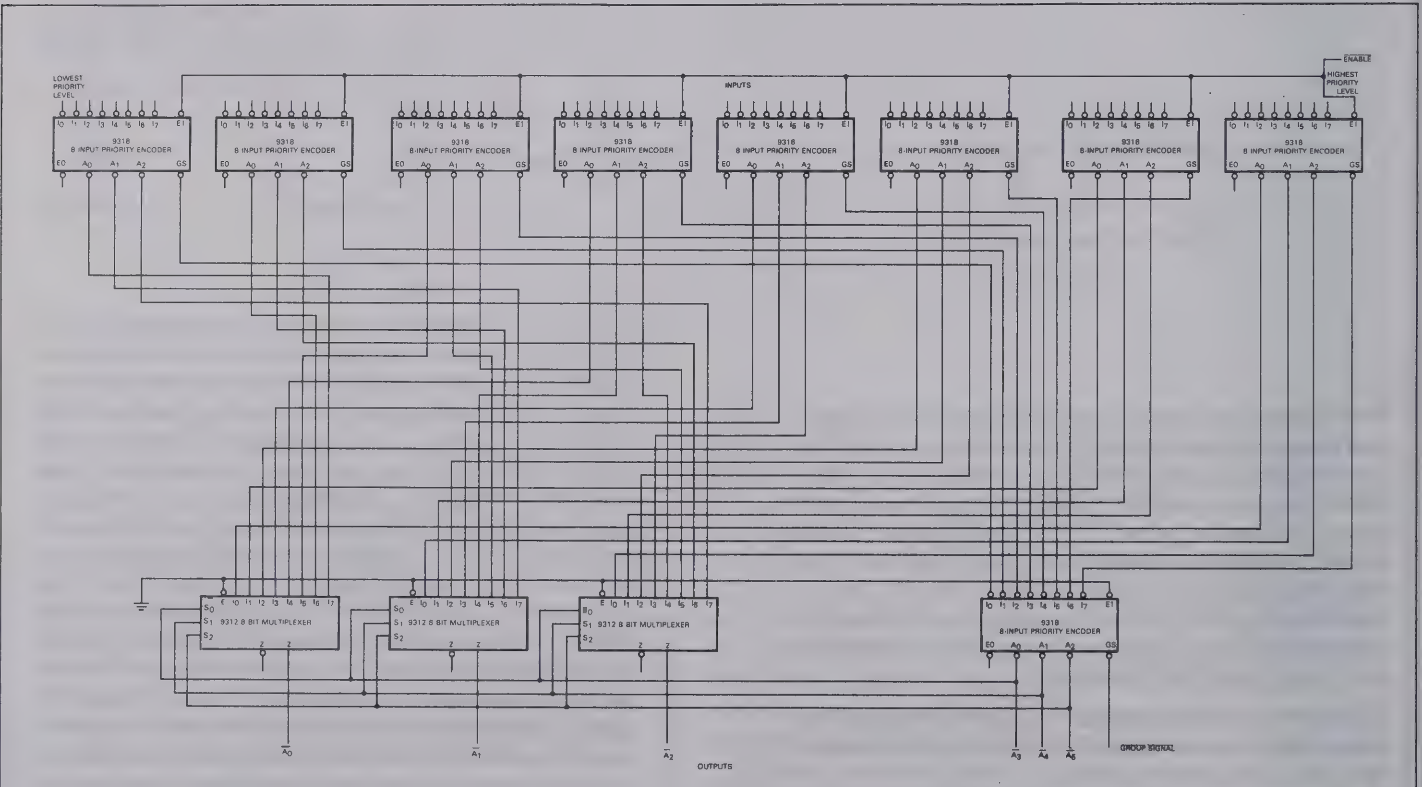
The disadvantage of series expansion is that (worst case) the Enable signals must ripple through every encoder before a valid output is provided.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

64-INPUT SERIES EXPANDED ENCODER


INPUT 64 DISCRETE SIGNALS
0 1 2 3 4 5 6 7
8
15
16
23
24
31
32
39
40
47
48
55
56
63
ENABLE
I0 I1 I2 I3 I4 I5 I6 I7 EI
9318
8 INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
9007
1/6 9016
2^0
2^1
2^2
OUTPUT
2^3
2^4
2^5
INTERRUPT

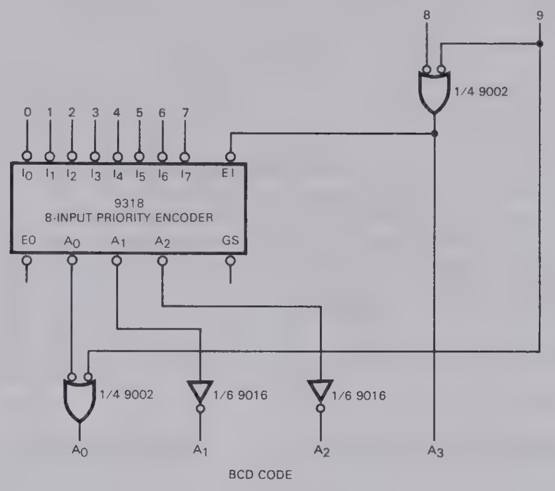
64-INPUT PARALLEL EXPANDED ENCODER


LOWEST PRIORITY LEVEL
INPUTS
ENABLE
HIGHEST PRIORITY LEVEL
I0 I1 I2 I3 I4 I5 I6 I7 EI
9318
8 INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
S0 S1 S2 E I0 I1 I2 I3 I4 I5 I6 I7
9312 8 BIT MULTIPLEXER
Z Z
Ā0
Ā1
Ā2
OUTPUTS
Ā3 Ā4 Ā5
GROUP SIGNAL

A fast expansion method (only 80 ns delay) is shown above. Each of the first level priority encoders operates independently. The group signals of each of the encoders are applied to another priority encoder that selects the highest order group signal and provides the most significant bits of the priority address. These bits are supplied as an address to three multiplexers which select the appropriate least significant bits. In this case the second level encoder must be a priority encoder, since more than one group signal can be active.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

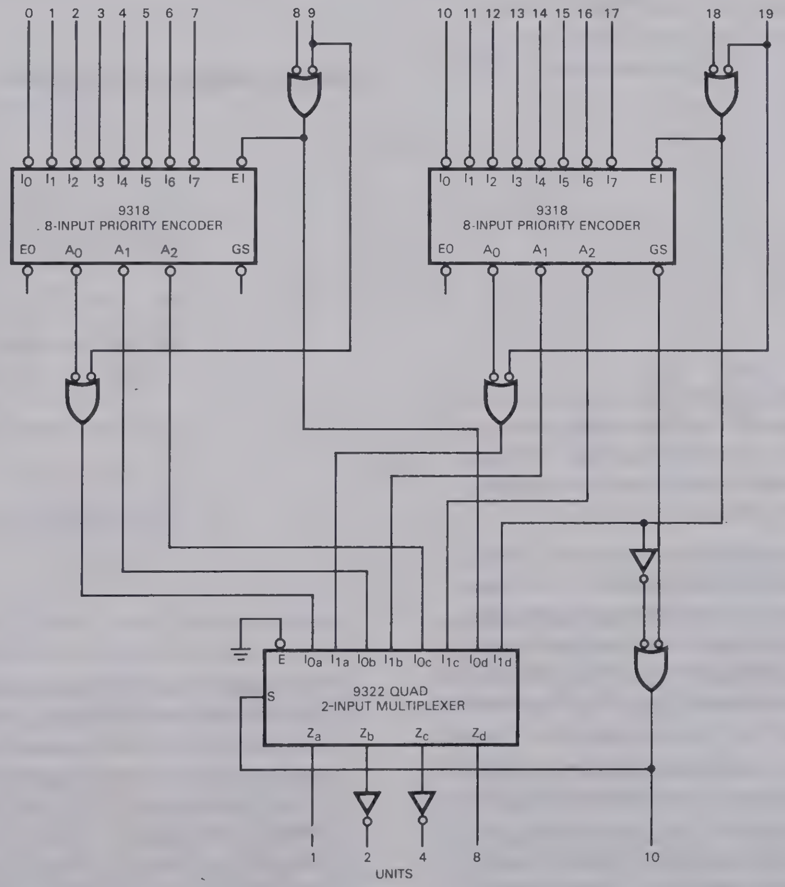
DECIMAL (10-INPUT) BCD 8421 ENCODER


8
9
1/4 9002
0 1 2 3 4 5 6 7
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
1/4 9002
1/6 9016
1/6 9016
A_0
A_1
A_2
A_3
BCD CODE

A 10-input, decimal-to-BCD 8421 code priority encoder is useful for a decimal keyboard or can be connected directly to a display decoder.

A 2-input NAND gate disables the 9318 when inputs $\overline{I}_8$ or $\overline{I}_9$ go Low and is used to produce the correct output code. When $\overline{I}_8$ or $\overline{I}_9$ are not Low, the encoder is enabled and encodes input $\overline{I}_0 - \overline{I}_7$ normally. This decimal encoder has active High outputs representing the highest order input. However, just inserting the two inverters in the A_0 and A_3 lines instead of the A_1 and A_2 lines provides active Low outputs.

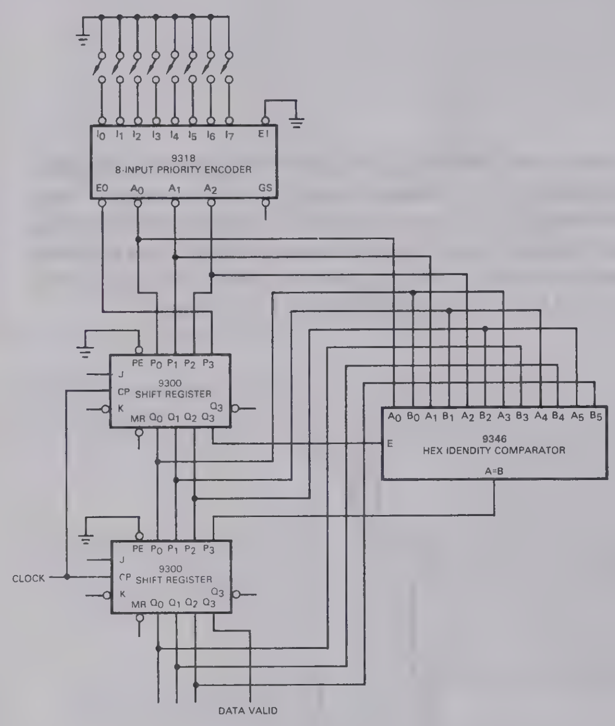
DECIMAL (20-INPUT) BCD 8421 ENCODER


0 1 2 3 4 5 6 7
8 9
10 11 12 13 14 15 16 17
18
19
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
E I_{0a} I_{1a} I_{0b} I_{1b} I_{0c} I_{1c} I_{0d} I_{1d}
S
9322 QUAD
2-INPUT MULTIPLEXER
Z_a Z_b Z_c Z_d
1
2
4
8
10
UNITS

A 20-input BCD encoder is formed by generating a Group Signal from the most significant decimal encoder, using this output as the 10s output and also using it to select the proper least significant digit through the 2-input multiplexer (9322).

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

SWITCH ENCODER

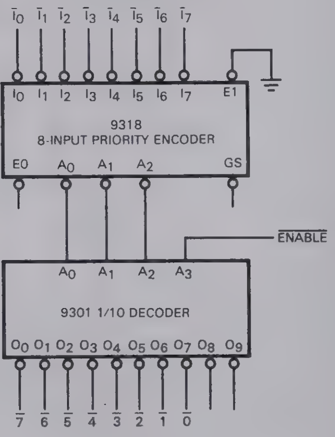

I0 I1 I2 I3 I4 I5 I6 I7 EI
9318
8-INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
PE P0 P1 P2 P3
J
9300
CP SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
A0 B0 A1 B1 A2 B2 A3 B3 A4 B4 A5 B5
E
9346
HEX IDENDITY COMPARATOR
A=B
CLOCK
DATA VALID

Switches or keys must be encoded for data entry into many digital systems. The 9318 is often most efficient for encoding groups of a moderate number of keys, from eight to 16. Other approaches, such as scanning, are more efficient for large keyboard arrays.

In this circuit, the output of the 9318 is sampled, stored, and compared over several clock periods. A Data Valid signal appears only after the 9318 outputs and the keys have stabilized. The clock rate supplied to the shift register must be adjusted to the bounce characteristics of the particular switch used so that all switch bounce is ignored.

Two 9300 shift registers are connected as four 2-bit shift registers so that codes from the 9318 are stored for two successive clock periods. All outputs are compared and Data Valid signal appears one clock pulse after identical output addresses have been clocked twice into the 9300 register. This insures that the output address is correct whenever the Data Valid signal is High.

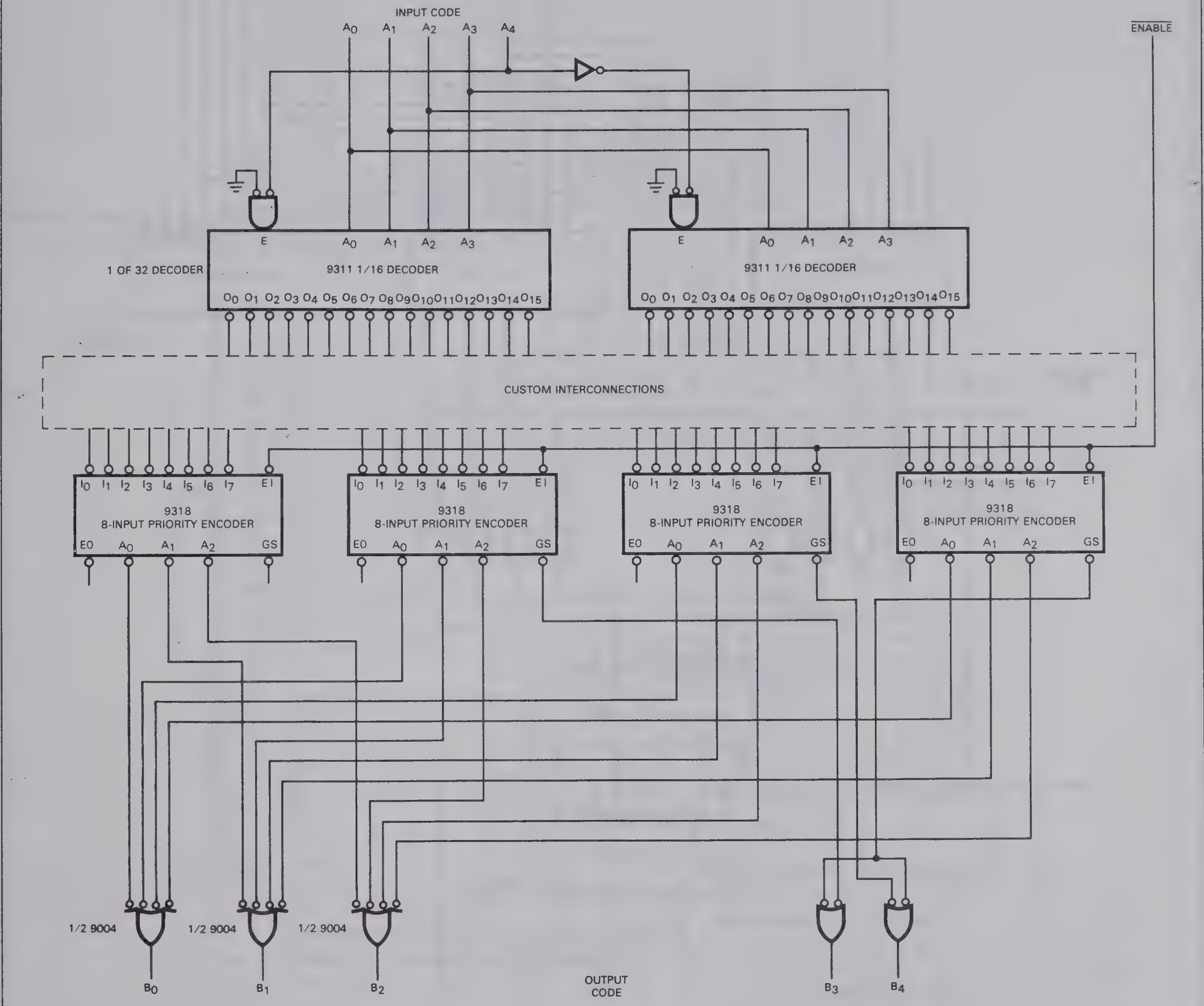
LINEAR PRIORITY ENCODER


Ī0 Ī1 Ī2 Ī3 Ī4 Ī5 Ī6 Ī7
I0 I1 I2 I3 I4 I5 I6 I7 EI
9318
8-INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
ENABLE
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
7̄ 6̄ 5̄ 4̄ 3̄ 2̄ 1̄ 0̄

The linear encoding network shown accepts eight active Low inputs and produces a single active Low output corresponding to the highest order input. The network consists of a 9318 to establish the address of the highest order input and a 9301 to decode this address and activate the appropriate output. This method offers a considerable package reduction over discrete linear priority networks and is easily expandable by adding more encoders and decoders. A 16-input encoding network requires only two 9318s, a 1-of-16 decoder (9311), and one gate package.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

CODE CONVERSION



INPUT CODE
A0 A1 A2 A3 A4
ENABLE
1 OF 32 DECODER
E A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
CUSTOM INTERCONNECTIONS
I0 I1 I2 I3 I4 I5 I6 I7 EI
9318
8-INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
1/2 9004
B0 B1 B2
OUTPUT CODE
B3 B4

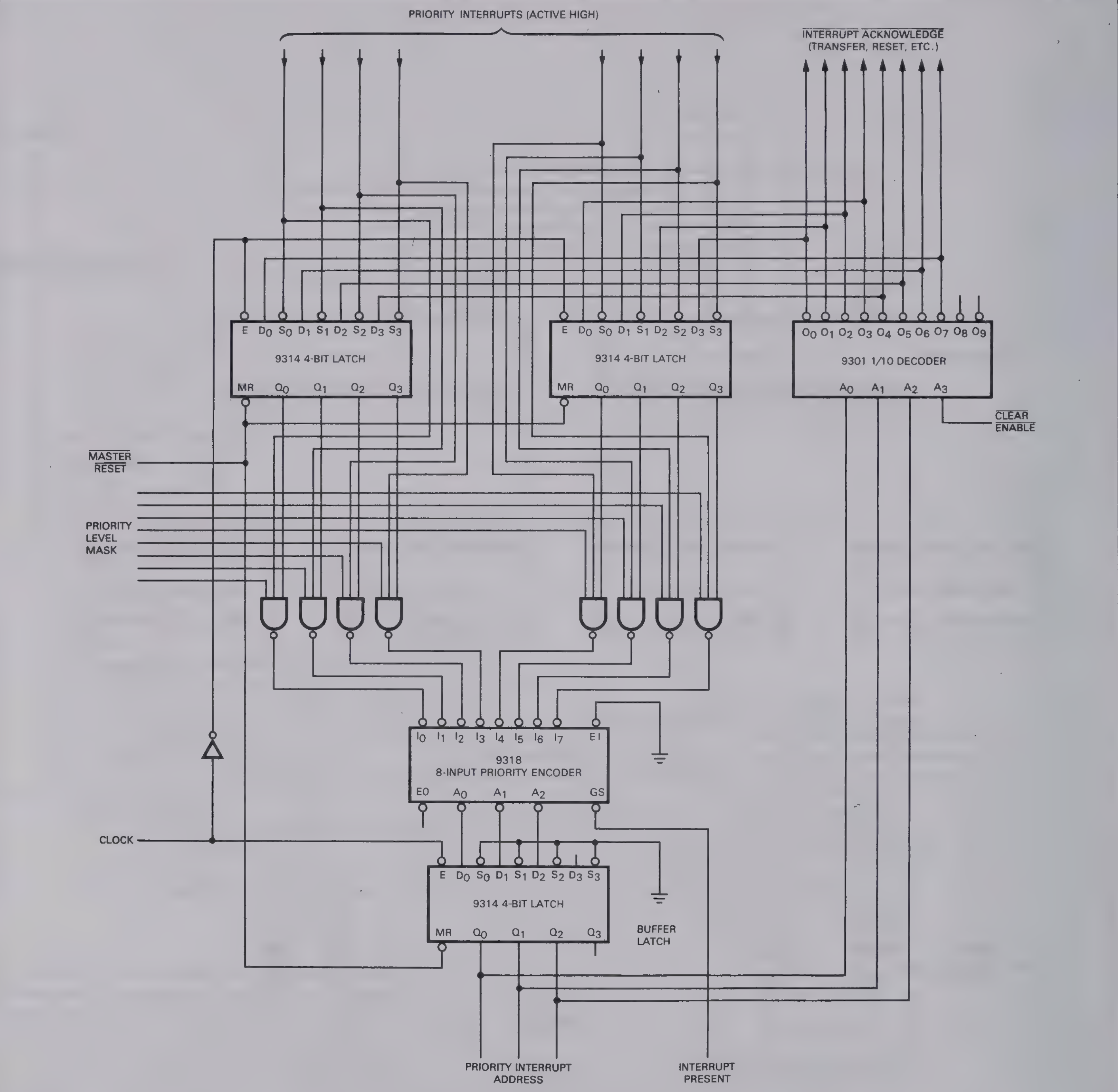
There are many instances where high speed, easily programmed conversions must be perfomed between binary codes, weighted or unweighted. A 5-bit arbitrary code converter utilizing the 9318 priority encoder is illustrated.

The 5-bit input code is applied to the 1-of-32 decoder (two 9311 1-of-16 decoders). Any state of the input code produces one Low output. This Low decoder output activates the particular encoder input that has been connected to the decoder output to produce the desired 5-bit output code. To cover all combinations of input codes, connections are made between every possible decoder output and the appropriate encoder inputs. Because only one output on the decoder is active for one input combination, it is not necessary for the 9318s to be interconnected in a priority configuration. Correspondingly, the group signal outputs of the first level do not occur simultaneously and can therefore be encoded with two gates.

The connections between decoder and priority encoders may be hard-wired or use switches, plug boards or multiplexing to facilitate a change of the code conversion.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

COMPUTER PRIORITY INTERRUPTS

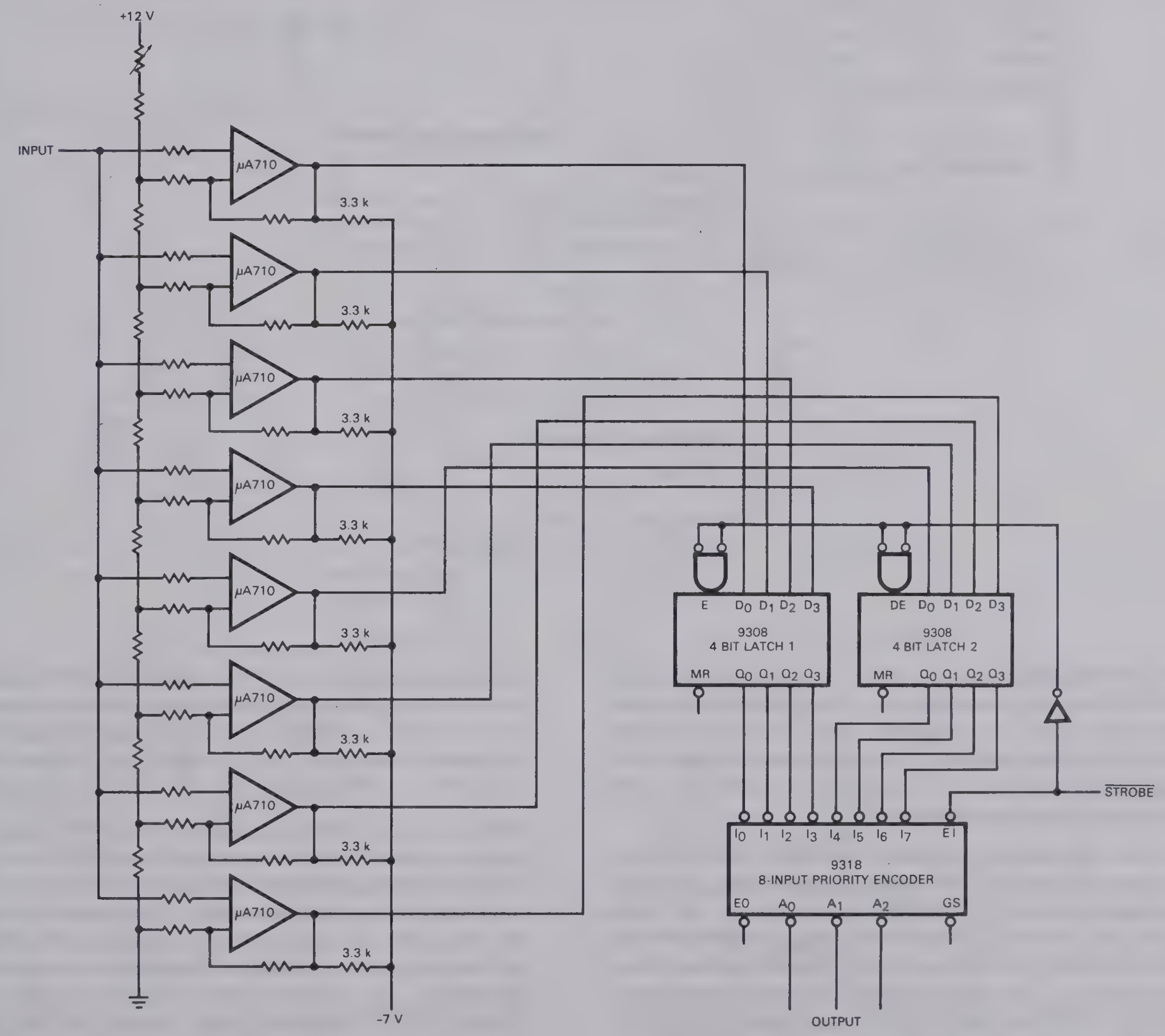

PRIORITY INTERRUPTS (ACTIVE HIGH)
INTERRUPT ACKNOWLEDGE (TRANSFER, RESET, ETC.)
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
9301 1/10 DECODER
A0 A1 A2 A3
CLEAR ENABLE
MASTER RESET
PRIORITY LEVEL MASK
I0 I1 I2 I3 I4 I5 I6 I7 EI
9318 8-INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
CLOCK
BUFFER LATCH
PRIORITY INTERRUPT ADDRESS
INTERRUPT PRESENT

The primary use of priority encoders is in priority interrupt systems, such as the system illustrated. Interrupts control the access of input and output units to the processing unit. By assigning priorities to the requesting units, simultaneous or multiple requests can be handled according to their significance, (priority). The interrupt circuitry in this system provides a signal to the processor indicating that an interrupt request has been made, and also generates the address of the highest order interrupt. All other lower order interrupt requests are ignored until the highest order interrupt is cleared.

When deactivated, the asynchronous interrupt lines hold the 9314 latches set. When a device initiates an interrupt request in the form of a High logic level, the priority encoder input is activated if that particular priority level is not masked. The masking is usually dynamically controlled from the program. The priority interrupt address is strobed from the encoder while the 9314 latches are disabled to insure that the 9318 input cannot change. After the interrupt has been acknowledged, the decoder is enabled to clear the priority input and send an acknowlegement signal to reset the interrupt request and initiate the transfer of data. The only requirement for the behavior of the interrupt signal is that it be reset (Low) before a new interrupt on that line can be recognized.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

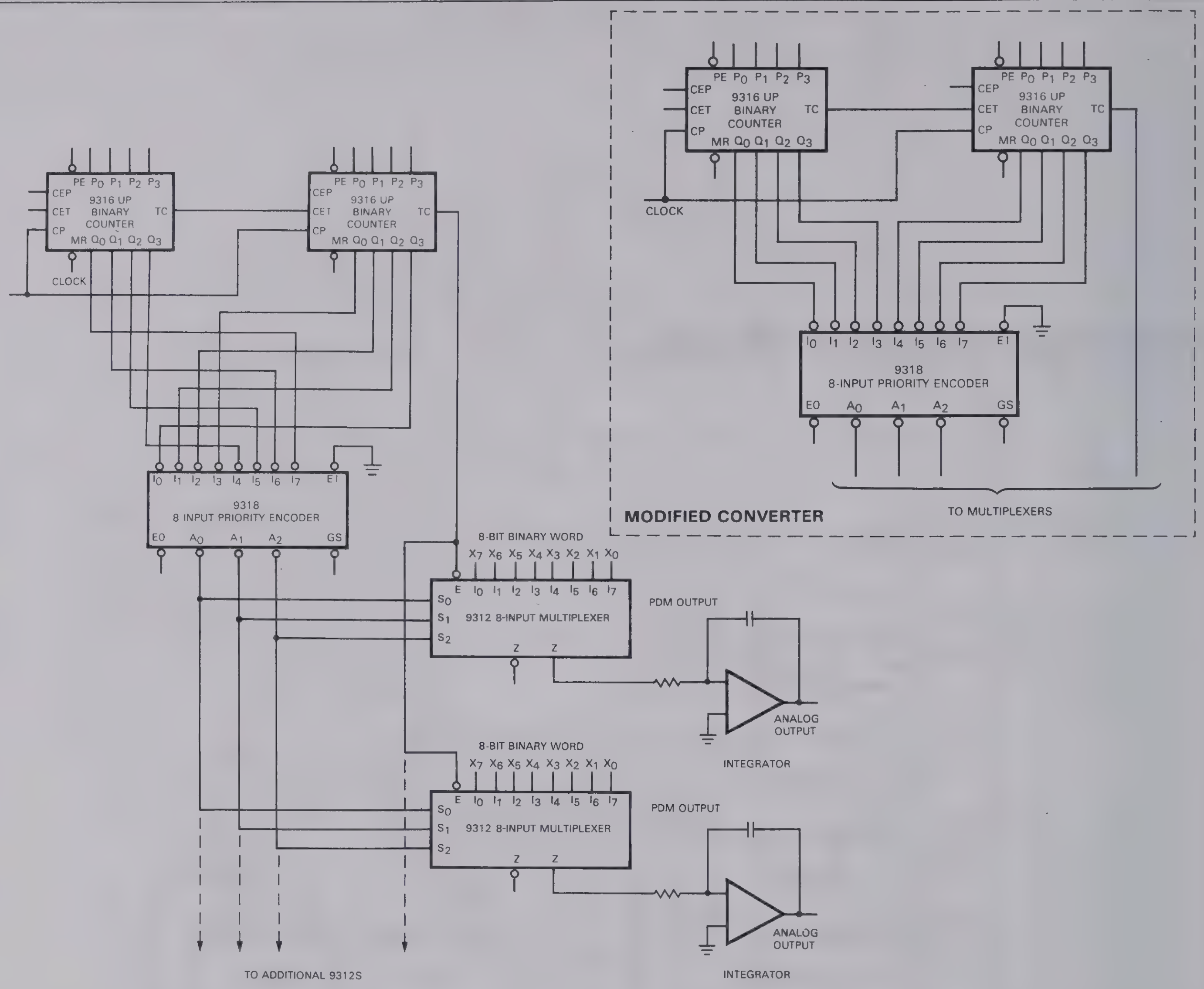
SIMULTANEOUS ANALOG TO DIGITAL CONVERSION


+12 V
INPUT
μA710
3.3 k
-7 V
E D0 D1 D2 D3
9308
4 BIT LATCH 1
MR Q0 Q1 Q2 Q3
DE D0 D1 D2 D3
9308
4 BIT LATCH 2
MR Q0 Q1 Q2 Q3
STROBE
I0 I1 I2 I3 I4 I5 I6 I7 EI
9318
8-INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
OUTPUT

Super fast A/D conversions can be performed by simultaneous multithreshold techniques. In this 3-bit binary A/D converter, the 9318 priority encoder provides all the encoding required. The analog signal voltage to be converted is connected to one input of each analog comparator. The other input of each comparator is connected to one of the seven reference voltages developed in the resistor voltage divider. The input signal is compared to the reference voltages and the comparators generate a High output when the input signal is greater than the reference voltage. The priority encoder detects the highest order zero, which corresponds to the lowest reference voltage that still exceeds the input voltage. To prevent transient incorrect encoder outputs while strobing the 9318 outputs, the μA710 outputs are stored in latches.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

DIGITAL-TO-ANALOG CONVERSIONS, BINARY RATE MULTIPLIER



9316 UP BINARY COUNTER
PE P0 P1 P2 P3
CEP
CET
CP
TC
MR Q0 Q1 Q2 Q3
CLOCK
9318 8 INPUT PRIORITY ENCODER
I0 I1 I2 I3 I4 I5 I6 I7 EI
EO A0 A1 A2 GS
8-BIT BINARY WORD
X7 X6 X5 X4 X3 X2 X1 X0
9312 8-INPUT MULTIPLEXER
E I0 I1 I2 I3 I4 I5 I6 I7
S0 S1 S2
Z Z
PDM OUTPUT
ANALOG OUTPUT
INTEGRATOR
TO ADDITIONAL 9312S
MODIFIED CONVERTER
TO MULTIPLEXERS

The 9318 can be used for digital-to-analog conversions. In this conversion technique, a rate multiplier is formed and the output is integrated. This technique is very economical for multiple D/A conversions because each additional channel of conversion requires only one multiplexer and one integrator.

In the converters illustrated, the eight bits of binary data are sampled (rate multipled) and during 256 clock periods converted to a PDM signal that is fed to an integrator producing the analog output. Each eight bits of digital input is independently sampled by an 8-input multiplexer. The 9318 supplies a code sequence to each multiplexer such that the most significant binary input is sampled for 50% of the count cycle, the next most significant input is sampled for 25% of the cycle, and so on. This sampling retains the weighting inherent in the binary code.

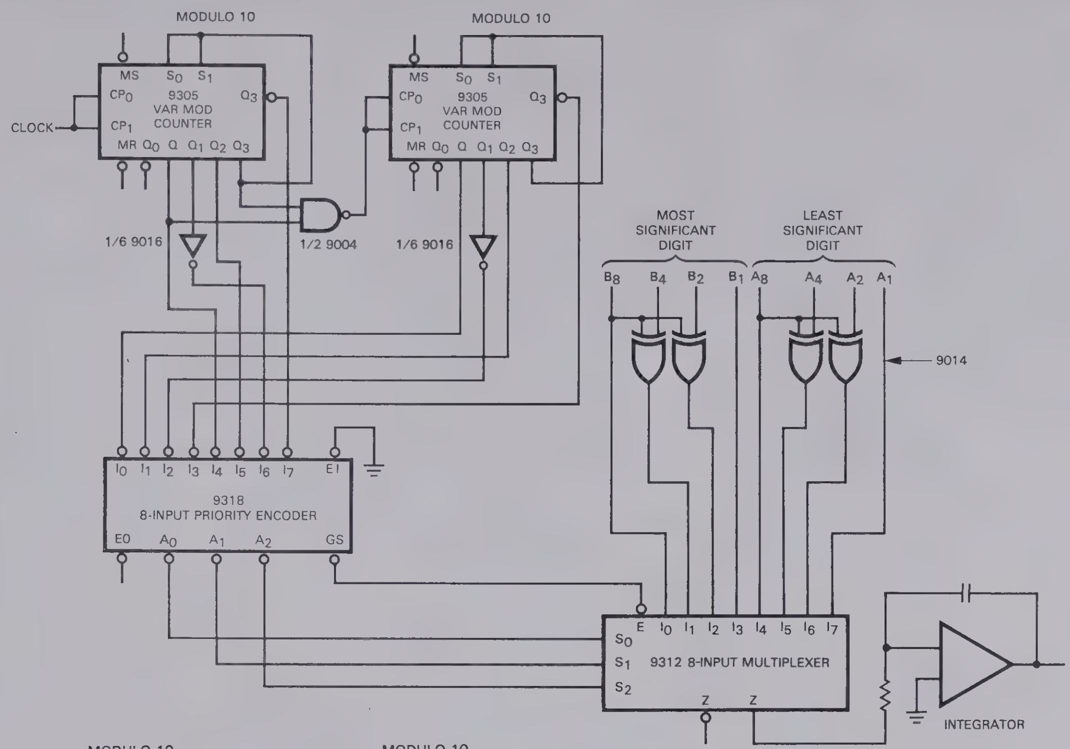
The converter above generates a well interlaced PDM signal equivalent to a narrow bandwidth that is easily integrated. The output can follow digital input data changes faster than in the other approach shown on the right. The output of the modified converter is not well interlaced and generates a wide bandwidth PDM signal. It therefore requires a longer integrating time constant, but it only has a maximum of eight logic changes per conversion cycle (vs. 256 changes) and is much less sensitive to switching delays, rise and fall times, etc.

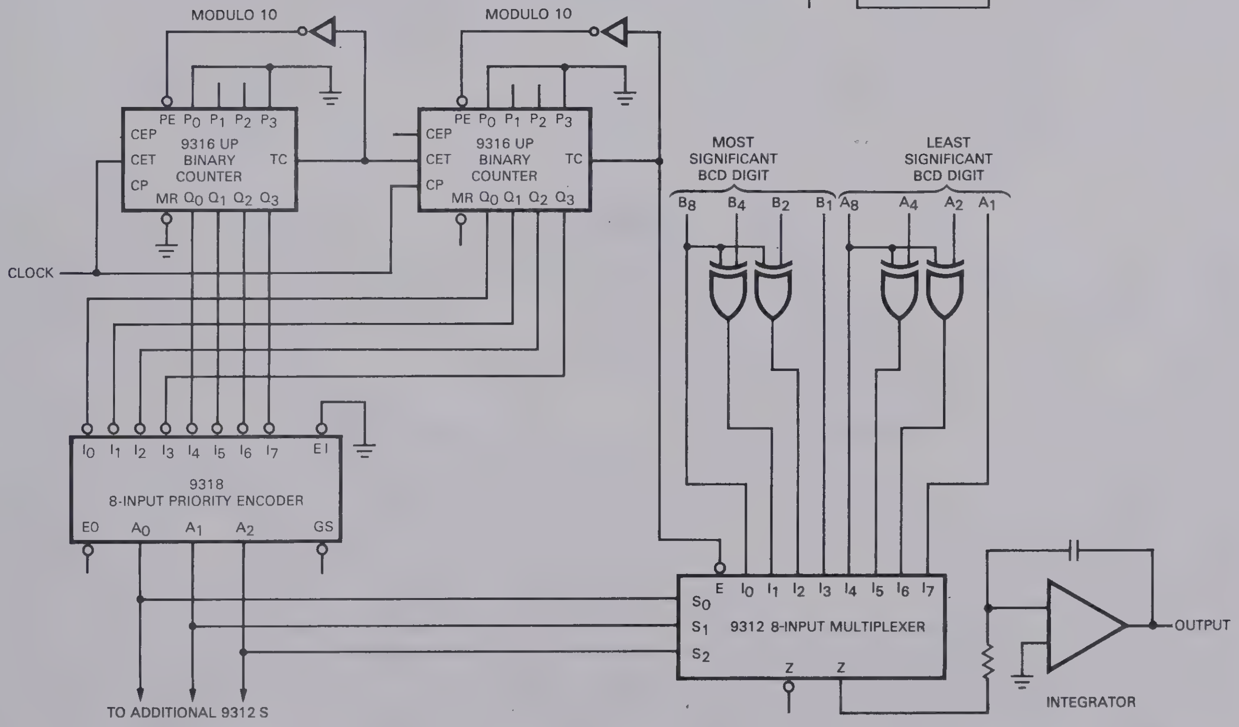
At high speeds, switching delays in the priority encoder and multiplexer introduce errors in the PDM output. A resynchronizing D flip-flop at each multiplexer output eliminates these cascaded delays. The maximum clock rate must allow enough time after the counter transition for the propagation delays in the priority encoder and multiplexer. The output of the multiplexer can be integrated or fed to integrating type devices such as panel meters, solenoids or motors.

For each additional channel, a multiplexer and integrator are required. To expand the conversion to more bits, the counter, priority encoder and multiplexer must be expanded. For example, a 16-bit converter requires a 16-bit counter, 16-input priority encoder and 16-bit multiplexer. As before, each additional channel only requires the addition of one multiplexer.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

DECIMAL D/A CONVERSION, BCD RATE MULTIPLIER

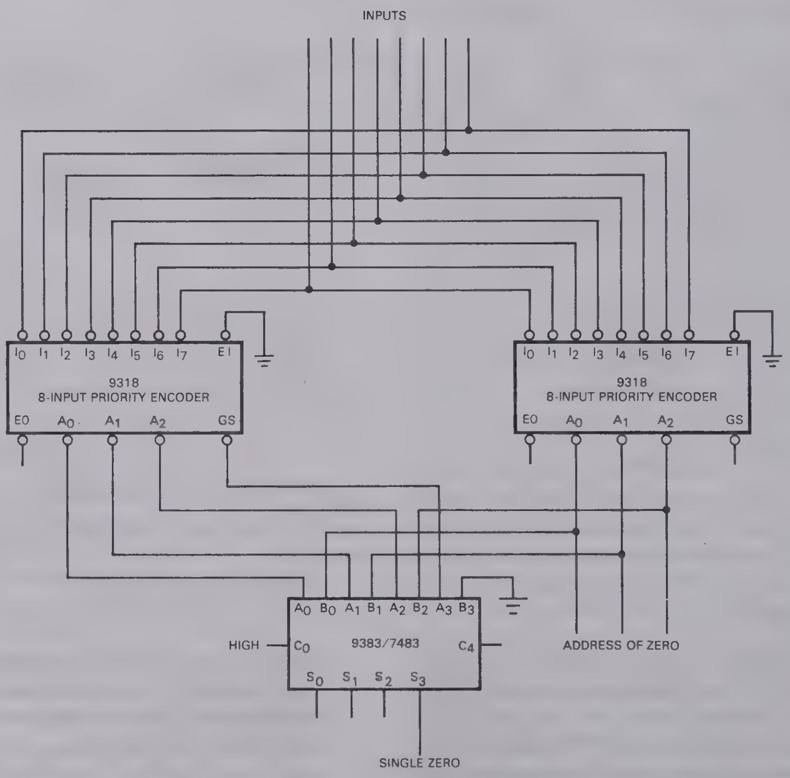

MODULO 10
MODULO 10
MS S_0 S_1
CP_0 9305 VAR MOD COUNTER Q_3
CP_1
MR Q_0 Q Q_1 Q_2 Q_3
CLOCK
1/6 9016
1/2 9004
1/6 9016
MOST SIGNIFICANT DIGIT
LEAST SIGNIFICANT DIGIT
B_8 B_4 B_2 B_1 A_8 A_4 A_2 A_1
9014
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318 8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
E I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
S_0 S_1 S_2
9312 8-INPUT MULTIPLEXER
Z Z
INTEGRATOR


MODULO 10
MODULO 10
PE P_0 P_1 P_2 P_3
CEP
9316 UP BINARY COUNTER
CET TC
CP
MR Q_0 Q_1 Q_2 Q_3
CLOCK
MOST SIGNIFICANT BCD DIGIT
LEAST SIGNIFICANT BCD DIGIT
B_8 B_4 B_2 B_1 A_8 A_4 A_2 A_1
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318 8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
E I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7
S_0 S_1 S_2
9312 8-INPUT MULTIPLEXER
Z Z
OUTPUT
INTEGRATOR
TO ADDITIONAL 9312 S

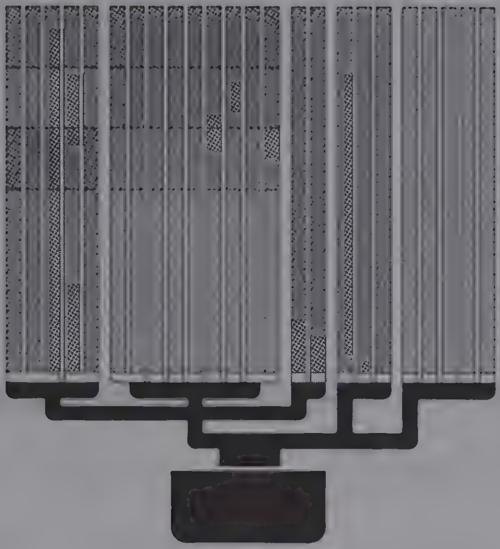
BCD 8421 code to analog signal conversion is similar to binary conversion. All the advantages of the binary D/A conversion are retained and only one additional gate package per channel is required. The extra package is needed to manipulate the BCD input data slightly so that correct sampling occurs. In the 2-digit BCD D/A converters or rate multipliers shown, a complete clock gate occurs every 100 clock pulses. The most significant digit is sampled 90% of the time and the least significant digit sampled 10% of the time. To obtain the correct weighting, the $A_1A_2A_4$ inputs are sampled respectively for one, two, and four sample times. The most significant input (A_8), is sampled two sample times alone and in addition is OR tied with the A_4 and A_2 inputs. Therefore, if the A_8 input is a one, the output is High for eight clock pulses. The PDM output is fed to an integrator to produce an analog output. The two decimal converters shown here differ in the same way as the binary converters shown in the preceding digital-to-analog conversions diagram. The first converter produces a well interlaced pattern and the second has less transitions per conversion cycle.

9318 8-INPUT PRIORITY ENCODER APPLICATIONS

SINGLE ZERO DETECTION


INPUTS
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
I_0 I_1 I_2 I_3 I_4 I_5 I_6 I_7 EI
9318
8-INPUT PRIORITY ENCODER
EO A_0 A_1 A_2 GS
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3
HIGH
C_0
9383/7483
C_4
S_0 S_1 S_2 S_3
ADDRESS OF ZERO
SINGLE ZERO

The circuit indicates both that a single zero is applied and which position it occupies. The outputs of each encoder are added so that the most significant bit of the sum is High only when a single active Low input signal is applied.

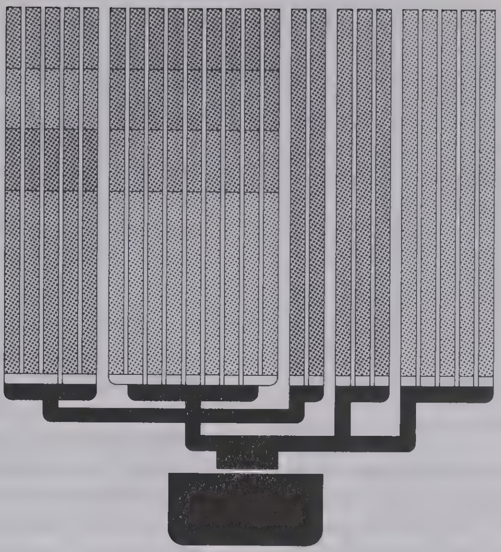


Operators

Adders/Subtractors
Multipliers/Dividers
Comparator Systems
Error Detection/Correction
Code Conversion

OPERATOR SELECTION GUIDE

Function	Device	# Of Devices Required For 16-Bit Function	Delay Over 16-Bits, ns (typ)	Power Dissipation, mW (typ)	Leads
Dual Full Adder	9304	8	150	150	16
Gated Full Adder	9380	16	150	105	14
2-Bit Full Adder	9382	8	150	175	14
4-Bit Ripple Carry Adder	9383	4	150	390	16
4-Bit Arithmetic Logic Unit	9340	4	38	425	24
4-Bit Arithmetic Logic Unit	9341	4	40	450	24
Carry Lookahead Generator	9342	1		180	16
4-Bit Arithmetic Logic Unit	93S41	4	20	500	24
Carry Lookahead Generator	93S42	1		260	16
2 x 4-Bit 2s Complement Multiplier	93S43	32	165	490	24
2 x 4-Bit Full Multiplier	9344	32	350	550	24
5-Bit Comparator	9324	4	72	200	16
5-Bit Comparator	93L24	4	100	50	16
12-Input Parity Checker/Generator	9348	2	40	235	16
9-Input Parity Checker/Generator	93S62	2	28	225	14



OPERATORS

INTRODUCTION

The term "operators" describes a broad category of combinatorial (non-memory) devices which perform logic operations such as AND, OR, Exclusive-OR, INVERT — arithmetic operations: Add, subtract, multiply, divide — and compare the magnitude of two operands or generate/check parity.

Because operators tend to be used in the heart of digital systems, they strongly influence system design and architecture. It is important to investigate the large number of alternate devices before settling on a system design. These devices represent compromises of speed, cost, part count, and connection complexity. The following points are some major design considerations.

SPEED — Slower systems usually require fewer and less expensive components and are less sensitive to noise. The system designer should always attempt to use all available time; perhaps by changing to serial architecture, or to incrementing counters, etc.

CODES — Binary arithmetic is simpler than decimal arithmetic. BCD and Excess-3 codes are preferred for decimal operation. Special codes (BCD and Excess-3 Gray) require extensive conversion before use in arithmetic operations.

NEGATIVE NUMBERS — For addition and subtraction, negative numbers are best represented as complements, one or twos complement in binary notation, nine or tens complement in decimal notation. The easiest to generate are one and nines complement; however, two and tens complements permit faster and simpler arithmetic. For multiplication and division, and for human interfacing (input/output) negative numbers are best represented in signed magnitude notation.

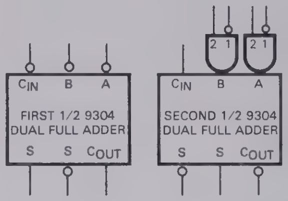
VERSATILITY — When several different operations are to be performed, a well designed Arithmetic Logic Unit (ALU) may be able to execute them in sequence. For example, an ALU can count by incrementing or decrementing a register, or it may be used to control a display multiplexer, etc.

Fairchild offers a broad range of operator devices in the TTL/MSI family, ranging in complexity from 2-bit adders to 4 x 2-bit multipliers and 4-bit Arithmetic Logic Units (ALUs).

The selection chart lists the operator devices and their more important parameters. For better comparison of the dissimilar devices, the through-delay is listed for typical operation on a 16-bit word.

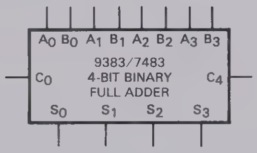
OPERATORS

9304 Dual Full Adder


C_{IN} B A
FIRST 1/2 9304 DUAL FULL ADDER
S S C_{OUT}
2 1 2 1
C_{IN} B A
SECOND 1/2 9304 DUAL FULL ADDER
S S C_{OUT}

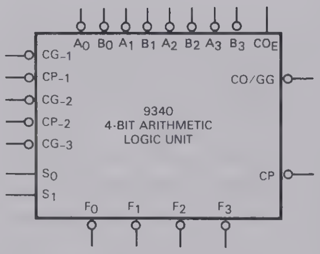
The most versatile full adder circuit is the 9304, two completely independent full adders. One of these adders has an additional set of opposite polarity inputs. The 9304 is used for serial addition and for addition of more than two variables.

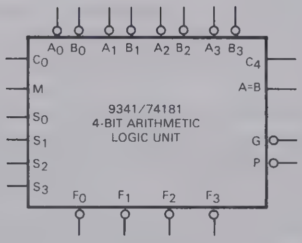
9383 4-Bit Ripple Carry Adder


A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3
C_0
9383/7483 4-BIT BINARY FULL ADDER
C_4
S_0 S_1 S_2 S_3

The 9383/7483 is four cascaded full adders. They add four bits of A with four bits of B plus a carry input, generating four sum bits and a carry output. There are no control inputs and the speed is limited by the internal ripple carry structure. However, this low cost 4-bit adder in a 16-lead package is useful in medium speed parallel binary systems and in character-serial BCD arithmetic.

9340, 9341 and 93S41 4-Bit Arithmetic Logic Units

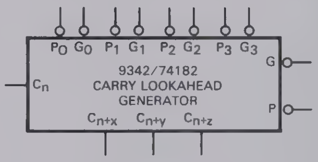

A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 CO_E
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
S_0
S_1
CO/GG
9340 4-BIT ARITHMETIC LOGIC UNIT
CP
F_0 F_1 F_2 F_3


A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3
C_0
M
S_0
S_1
S_2
S_3
C_4
A=B
9341/74181 4-BIT ARITHMETIC LOGIC UNIT
G
P
F_0 F_1 F_2 F_3

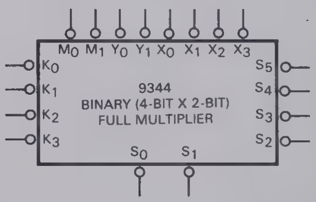
The 9340 and the 9341/74181 are 4-bit Arithmetic Logic Units of much higher complexity, versatility, and speed than the adders mentioned so far. They come in 24-lead packages and are used in parallel, high speed binary systems. They accept four bits of one operand (A) and four bits of a second operand (B) and generate four outputs (F). Operation is controlled by Mode and Select inputs.

The 9340 can perform two arithmetic functions (add, subtract) and two logic functions, while the 9341/74181 can perform 16 different arithmetic functions and all 16 possible logic functions of the two variables. In their arithmetic mode both the 9340 and the 9341/74181 use fast carry lookahead circuitry, but only the 9340 has sufficient carry inputs to extend the carry lookahead over 16 bits. The 9341/74181 requires an auxiliary circuit, the 9342/74182, to perform fast carry lookahead arithmetic over more than eight bits.

9342,93S42 Carry Lookahead Generator

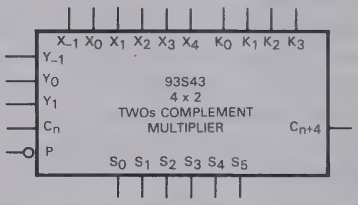

P_0 G_0 P_1 G_1 P_2 G_2 P_3 G_3
C_n
9342/74182 CARRY LOOKAHEAD GENERATOR
G
P
C_{n+x} C_{n+y} C_{n+z}

9344 Combinatorial Multiplier


M_0 M_1 Y_0 Y_1 X_0 X_1 X_2 X_3
K_0
K_1
K_2
K_3
9344 BINARY (4-BIT X 2-BIT) FULL MULTIPLIER
S_5
S_4
S_3
S_2
S_0 S_1

The 9344 is an expandable 4 x 2-bit combinatorial multiplier building block with inputs and outputs in signed magnitude notation.

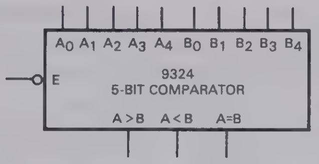
93S43 Combinatorial Multiplier


X_{-1} X_0 X_1 X_2 X_3 X_4 K_0 K_1 K_2 K_3
Y_{-1}
Y_0
Y_1
C_n
P
93S43 4 x 2 TWOs COMPLEMENT MULTIPLIER
C_{n+4}
S_0 S_1 S_2 S_3 S_4 S_5

The 93S43 is a super-fast expandable 4 x 2-bit combinatorial multiplier building block with inputs and outputs in twos complement notation. Both the 9344 and 93S43 are used for fast multiplication in systems which can not tolerate the delay of conventional shift-and-add circuits. Two binary factors of any length can be multiplied using nothing but a number of 9344s or 93S43s. This is useful in fast real-time applications such as Fast Fourier Transform.

OPERATORS

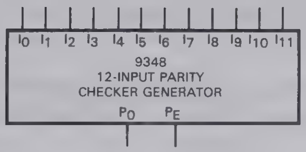
9324,93L24 Comparator



The 9324 is a 5-bit *magnitude* comparator that can also be considered an expandable 4-bit *magnitude* comparator. It accepts two sets (A and B) of five bits each and generates three mutually exclusive outputs A > B, A < B, A = B.

The simpler function of *identity* comparison (detecting whether or not two numbers are identical) is faster and more economical with Exclusive-OR circuits.

9348 Parity Checker Generator



The 9348 is a 12-input parity checker or generator circuit used in error detecting and correcting applications on parallel data.

TERMINAL CONFUSION *or* HOW CAN TWO STATES LEAD TO SO MANY TERMS?

The signals used in digital systems are described in several different and sometimes confusing terms. A logic signal can be either ACTIVE (=TRUE) or NOT ACTIVE (NOT TRUE = FALSE) Digital circuits, on the other hand are defined for voltage levels that are either HIGH (more positive) or LOW (less positive or more negative). Either of these levels can be considered ACTIVE (TRUE), the opposite level is then NOT ACTIVE (FALSE).

MIL Std 806 has established a clear symbology: The HIGH level is considered ACTIVE unless a small circle ("bubble") at the input or output describes the opposite assignment (LOW = ACTIVE).

In non-arithmetic circuits the symbols "0" and "1" are unnecessary and confusing because some people think that a one implies a HIGH level, others think of it as an ACTIVE (TRUE) signal, and some mistakenly think that it must mean both ACTIVE and HIGH.

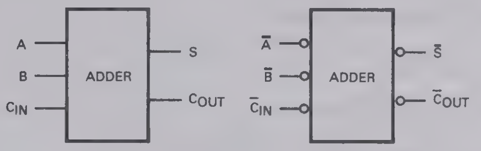
Therefore, this book generally does not use zero and one, but uses the terms ACTIVE and NOT ACTIVE for systems descriptions and the terms H and L for circuit descriptions and truth tables.

In arithmetic (binary and BCD) systems the terms zero and one can not be avoided, since they have a mathematical significance. They have to be related to the logic terms in a consistent and unambiguous way.

Arithmetic 1 = ACTIVE = TRUE
Arithmetic 0 = NOT ACTIVE = NOT TRUE = FALSE

The rules of Mil Std 806B are then used to describe whether a HIGH level means a one (active High, no bubble) or whether a LOW level means a one (active Low, with a bubble at the input or output of the logic symbol).

FUNCTIONS



Active High Logic Function					Electrical Function					Active Low Logic Function				
A	B	C_{IN}	S	C_{OUT}	A	B	C_{IN}	S	C_{OUT}	$\overline{A}$	$\overline{B}$	$\overline{C}_{IN}$	$\overline{S}$	$\overline{C}_{OUT}$
0	0	0	0	0	L	L	L	L	L	1	1	1	1	1
1	0	0	1	0	H	L	L	H	L	0	1	1	0	1
0	1	0	1	0	L	H	L	H	L	1	0	1	0	1
1	1	0	0	1	H	H	L	L	H	1	1	1	1	0
0	0	1	1	0	L	L	H	H	L	1	1	0	0	1
1	0	1	0	1	H	L	H	L	H	0	1	0	1	0
0	1	1	0	1	L	H	H	L	H	1	0	0	1	0
1	1	1	1	1	H	H	H	H	H	0	0	0	0	0

A full adder produces sum and carry outputs as a function of the three inputs A, B, and C. The center truth table above describes the *electrical* function in terms of High and Low. The two logic truth tables and the two logic symbols describe this circuit in terms of *either* active High or active Low *logic* levels. Any logic network which perfoms binary addition or subtraction can be described in terms of active High as well as in terms of active Low inputs and outputs.

Such *equivalence* is a basic feature of adder structures and is true regardless of the number of bits and the method of carry propagation. It applies to a single full adder as well as to a complex ALU system.

CARRY SIGNALS IN PARALLEL BINARY ADDERS

High speed digital systems perform addition and subtraction on parallel words of typically 8 to 64 bits. The result of an addition or subtraction at any bit position, however, depends not only on the two operand bits in that position, but also on the less significant operand bits. More specifically, the result depends on the carry from the less significant bit positions.

Ripple Carry

In the simplest scheme, each position receives a potential carry input from the less significant position and passes a potential carry on to the more significant position. Thus the worst case delay for the addition of two n-bit numbers is n-1 carry delays plus one sum delay. This technique is used with simple adders like the 9304 and the 9383/7483 4-bit ripple carry adder. It uses a minimum of hardware, but it is rather slow.

Carry Lookahead

Addition and subtraction can be made much faster if more logic is used at each bit position to anticipate the carry into this position instead of waiting for a ripple carry to propagate through all the lower positions. An adder constructed with carry anticipation is called a "carry lookahead adder".

The carry into position 0 is C_0

The carry into position 1 is $C_1 = A_0 \bullet B_0 + C_0 (A_0 + B_0)$

The carry into position 2 is $C_2 = A_1 \bullet B_1 + C_1 (A_1 + B_1)$

If the two auxiliary functions & and V are defined

$$\&_i = A_iB_i$$

$$V_i = A_i + B_i$$

then the carry equations are:

$C_1 = \&_0 + V_0C_0$

$C_2 = \&_1 + V_1 (\&_0 + V_0C_0)$

$C_3 = \&_2 + V_2 (\&_1 + V_1\&_0 + V_1V_0C_0)$

or, in general terms:

$C_{i+1} = \&_i + V_i\&_{i-1} + V_iV_{i-1}\&_{i-2} + V_iV_{i-1}V_{i-2}\&_{i-3} + \ldots$

The anticipated carry into any position can thus be generated in two gate delays (counting AND/OR/INVERT as one gate delay), one gate delay to generate all the & and V functions, and a second gate delay to generate the anticipated carry. The sum/difference outputs are generated in one additional delay for a total of three gate delays, *independent of word length.* The auxiliary functions & and V can be interpreted as

& = Carry Generate – AB generates a carry, independent of any incoming carry

V = Carry Propagate – A + B pass on an incoming carry

This "brute force" carry lookahead scheme is conceptually simple, but, due to the large number of interconnections and the heavy loading of the & and V functions, becomes impractical as the word length increases beyond five or six bits.

The same concept, however, can be applied on a higher level by dividing the word into practical blocks of 4-bit lengths, using carry lookahead within each block, generating new auxiliary functions G, Carry Generate and P, Carry Propagate which refer to the whole block. G is obviously the carry out of the most significant position of the block. P is defined as Carry Propagate through the block i.e., P is True if a carry into the block would result in a carry out of the block. For a block size of four bits (the 9340 and the 9341/74181)

$G = \&_3 + V_3\&_2 + V_3V_2\&_1 + V_3V_2V_1\&_0$

$P = V_3V_2V_1V_0$

Neither of these functions is affected by the incoming carry; they will therefore be stable within two gate delays and can be used to supply carry information to the more significant blocks. The carry into block n is:

$C_n = G_{n-1} + P_{n-1}G_{n-2} + P_{n-1}P_{n-2}G_{n-3} + \ldots$

This carry in signal is used in the internal carry lookahead structure:

$C_0 = C_n$

$C_1 = \&_0 + V_0C_n$

$C_2 = \&_1 + V_1\&_0 + V_1V_0C_n$

$C_3 = \&_2 + V_2\&_1 + V_2V_1\&_0 + V_2V_1V_0C_n$

The TTL MSI carry lookahead arithmetic logic units, the 9340 and the 9341/74181 use this 2-level carry lookahead, but because of connection differences, they differ in partitioning. The 9340 incorporates the carry in logic in the adder device, but limits it to inputs from three less significant blocks. This gives full carry lookahead over 16 bits, using four 9340 packages.

The 9341 has more logic flexibility, which requires three additional mode control inputs. It can not therefore, contain any carry in logic. It is contained in a separate device, the 9342/74182. Only one 9342 is needed to achieve full carry lookahead over 16 bits.

NUMBER REPRESENTATION

All presently available TTL/MSI adders and ALUs work on binary numbers. Operation in other number systems, such as BCD, Excess 3, etc. is achieved by additional logic and/or additional cycles through the binary adder.

There is only one way to represent positive binary numbers, but negative binary numbers can be represented in three ways.

- Sign Magnitude — The most significant bit indicates the sign (0 = positive, 1 = negative). The remaining bits indicate the magnitude, represented as a positive number.

Sign				LSB	
0	1	1	0	1	= +13
1	1	1	0	1	= –13

  This representation is convenient for multiplication and division, and may be desirable for human-oriented input and output, but, for addition and subtraction, it is inconvenient and rarely used.

- One Complement — Negative numbers are bit inversions of their positive equivalents. The most significant bit indicates the sign (0 = positive, 1 = negative). Thus –A is actually represented as $2^n - A - 1$. The one complement is very easy to form, but it has several drawbacks, notably a double representation for Zero (all Ones or all Zeros)

- Two Complement — This is the most common representation. It is more difficult to generate than one complement, but it simplifies addition and subtraction. The two complement is generated by inverting each bit of the positive number and adding one to the LSB.

Sign				LSB	
0	1	1	0	1	= +13
1	0	0	1	1	= –13

  Thus an n-bit word can represent the range from $+(2^{n-1} - 1)$ to $-(2^{n-1})$

A 4-bit word can represent the range from 0111 = +7 to 1000 = –8.

ADDITION AND SUBTRACTION OF BINARY NUMBERS

Addition of positive numbers is straightforward, but a carry into the sign bit must be prevented and interpreted as overflow. When two negative numbers or a negative and a positive number are added, the operation depends on the negative number representation. In two complement methods, addition is straightforward, but it must include the sign bit. Any carry out of the sign position is simply ignored.

$$\begin{array}{rr} +14 & 01110 \\ -7 & 11001 \\ \hline +7 & 00111 \end{array} \qquad \begin{array}{rr} +7 & 00111 \\ -14 & 10010 \\ \hline -7 & 11001 \end{array} \qquad \begin{array}{rr} -4 & 11100 \\ -3 & 11101 \\ \hline -7 & 11001 \end{array}$$

If one complement notation is used, the operation is similar, but the carry out of the sign bit must be used as a carry input to the least significant bit (LSB). This is commonly called "end-around carry".

$$\begin{array}{rr} +14 & 01110 \\ -7 & 11000 \\ \hline & 00110 \\ + & 1 \\ \hline +7 & 00111 \end{array} \qquad \begin{array}{rr} +7 & 00111 \\ -14 & 10001 \\ \hline -7 & 11000 \end{array} \qquad \begin{array}{rr} -4 & 11011 \\ -3 & 11100 \\ \hline & 10111 \\ + & 1 \\ \hline -7 & 11000 \end{array}$$

In two complement subtraction the arithmetic is performed by inverting; i.e., one complement, the subtrahend and adding, and by forcing a carry into the least significant bit (LSB).

$$\begin{array}{rr} +14 & 01110 \\ -(+7) & -00111 \\ & 01110 \\ & +11000 \\ + & 1 \\ \hline +7 & 00111 \end{array} \qquad \begin{array}{rr} +7 & 00111 \\ -(+14) & -01110 \\ & 00111 \\ & +10001 \\ + & 1 \\ \hline -7 & 11001 \end{array} \qquad \begin{array}{rr} -6 & 11010 \\ -(+8) & -01000 \\ & 11010 \\ & +10111 \\ + & 1 \\ \hline -14 & 10010 \end{array}$$

In one complement methods, subtraction is performed by inverting, i.e., one complement, the subtrahend and adding, using the Carry Out of the sign position as carry input to the LSB (end-around carry).

$$\begin{array}{rr} +14 & 01110 \\ -(+7) & -00111 \\ & 01110 \\ & +11000 \\ \hline & 00110 \\ + & 1 \\ \hline +7 & 00111 \end{array} \qquad \begin{array}{rr} +7 & 00111 \\ -(+14) & -01110 \\ & 00111 \\ & +10001 \\ \hline -7 & 11000 \end{array} \qquad \begin{array}{rr} -6 & 11001 \\ -(+8) & -01000 \\ & 11001 \\ & +10111 \\ \hline & 10000 \\ + & 1 \\ \hline -14 & 10001 \end{array}$$

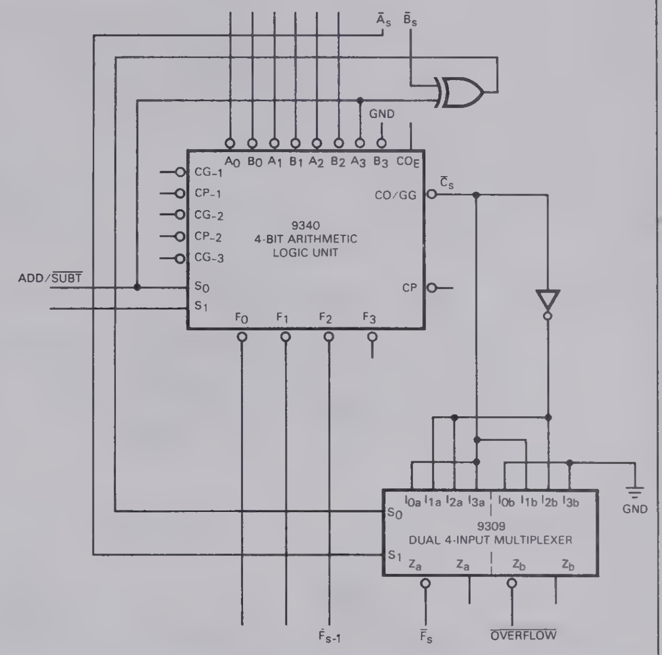
It is interesting to note that the Carry Out of the sign position occurs when the result does *not* change sign; *no* carry occurs when the sign changes, implying a "borrow".

OVERFLOW

Adding two numbers of the same sign or subtracting two numbers of opposite sign might generate a result which cannot be represented by the given word length. This is overflow. It must be detected and used to initiate some corrective routine. Overflow occurs when the Carry Out of the sign position differs from the Carry In to the sign position.

$$\text{OVERFLOW} = C_s \oplus C_{s+1}$$

Overflow Generation



When the sign is in the most significant position of a 9340 or 9341 ALU, the carry signal into this position is not directly available but must be regenerated in one of several ways. The simplest method is to use the equation

$$C_s = S_s \oplus A_s \oplus B_s \text{ and}$$

$$\text{OVERFLOW} = S_s \oplus A_s \oplus B_s \oplus C_{s+1}$$

An improved method of generating the overflow signal requires only two gate delays after the Carry Out is available. This allows the overflow to appear at about the same time as the sum. The sign bit is processed completely outside the ALU. The last bit of the ALU where the sign bit would normally be, is set with $\overline{B}_3$ = Low and $\overline{A}_3$ = Add/Subt. This will force a permanent Carry Propagate condition on bit F_3 so that the Carry Out of the ALU will be the Carry Out of the bit F_2 position. The Carry Out of the ALU is then C_s, the Carry In to the sign bit. The equation for the overflow can be written as:

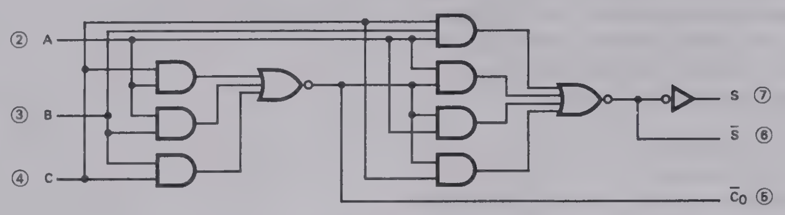
$$\text{OVERFLOW} = A_s \bullet \bar{C}_s \bullet (\bar{B}_s \oplus \text{Add}/\overline{\text{Subt}}) + \bar{A}_s \bullet C_s \bullet (B_s \oplus \text{Add}/\overline{\text{Subt}})$$

Both the overflow and the final sum sign can be produced by an 9309 dual 4-input multiplexer used as a random function generator as shown. Since the C_s signal goes into the data input of the multiplexer, the overflow is generated only two gate delays later; the final sum digit is produced at approximately the same time as the other sum digits since the logic has to wait for the C_s signal before the sum bit can be generated.

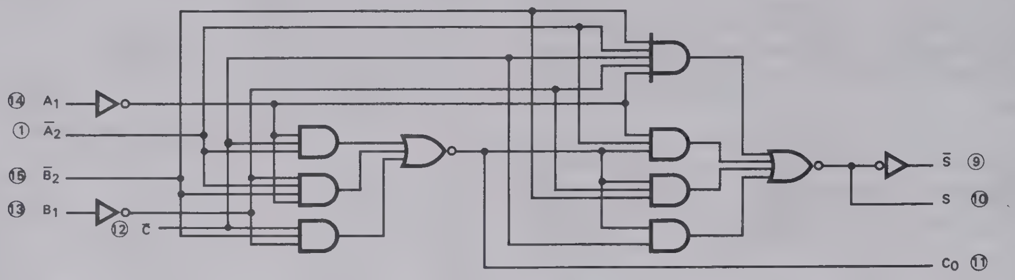
ADDERS/SUBTRACTORS

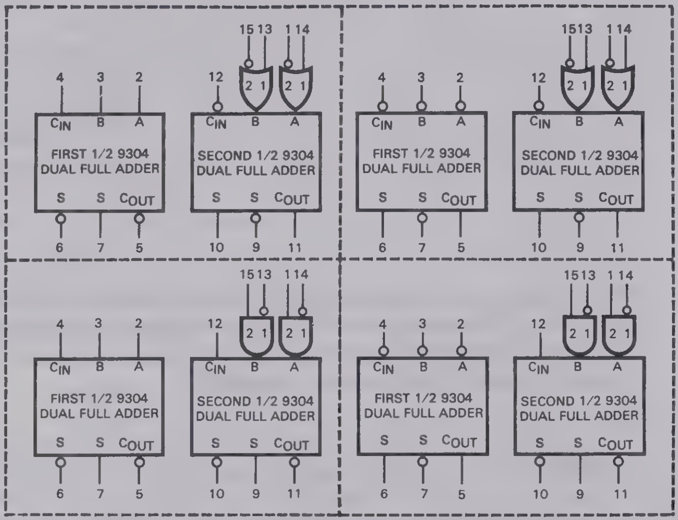
9304 DUAL FULL ADDER DESCRIPTION AND OPERATION

FIRST HALF



SECOND HALF




4 3 2
CIN B A
FIRST 1/2 9304 DUAL FULL ADDER
S S COUT
6 7 5
15 13 1 14
12
2 1 2 1
SECOND 1/2 9304 DUAL FULL ADDER
10 9 11

FIRST HALF

INPUT LEADS			OUTPUT LEADS		
4	3	2	5	6	7
L	L	L	H	H	L
L	L	H	H	L	H
L	H	L	H	L	H
L	H	H	L	H	L
H	L	L	H	L	H
H	L	H	L	H	L
H	H	L	L	H	L
H	H	H	L	L	H

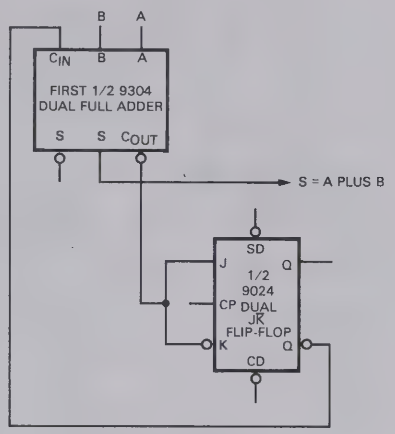
SECOND HALF

INPUT LEADS					OUTPUT LEADS		
12	13	14	15	1	11	10	9
L	L	L	L	L	H	H	L
L	L	L	L	H	H	L	H
L	L	L	H	L	H	L	H
L	L	L	H	H	L	H	L
L	L	H	L	L	H	H	L
L	L	H	L	H	H	H	L
L	L	H	H	L	H	L	H
L	L	H	H	H	H	L	H
L	H	L	L	L	H	H	L
L	H	L	L	H	H	L	H
L	H	L	H	L	H	H	L
L	H	L	H	H	H	L	H
L	H	H	L	L	H	H	L
L	H	H	L	H	H	H	L
L	H	H	H	L	H	H	L
L	H	H	H	H	H	H	L
H	L	L	L	L	H	L	H
H	L	L	L	H	L	H	L
H	L	L	H	L	L	H	L
H	L	L	H	H	L	L	H
H	L	H	L	L	H	L	H
H	L	H	L	H	H	L	H
H	L	H	H	L	L	H	L
H	L	H	H	H	L	H	L
H	H	L	L	L	H	L	H
H	H	L	L	H	L	H	L
H	H	L	H	L	H	L	H
H	H	L	H	H	L	H	L
H	H	H	L	L	H	L	H
H	H	H	L	H	H	L	H
H	H	H	H	L	H	L	H
H	H	H	H	H	H	L	H

The 9304 dual full adder contains two independent full adders with both polarities of the Sum output brought out. The Carry Out signal is produced in one AND/OR/INVERT gate delay, while the Sum requires one more AOI delay. This minimizes ripple carry delays, which are usually the critical parameter in a parallel adder.

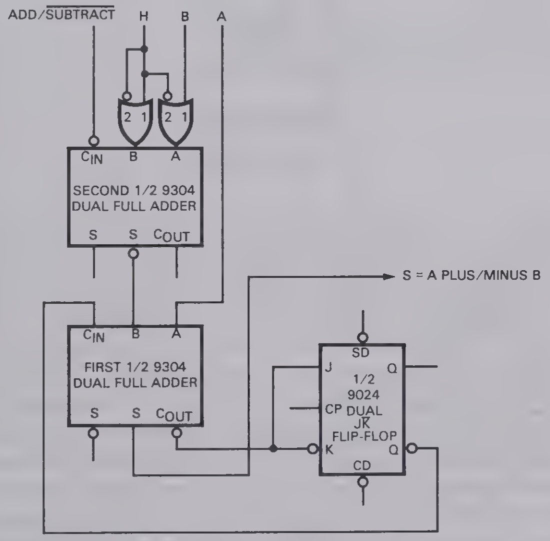
The second adder has dual inputs (one active High, one active Low) for A as well as B. This feature greatly enhances the versatility of the device. Because full adder functions are symmetrical the 9304 can be described in terms of active High as well as active Low inputs and outputs. Both logic representations and all four methods of using the 9304 dual full adder are shown here. Note that inputs A, B, and C can be arbitrarily interchanged, without affecting the operation of the device. This can simplify printed circuit board layout.

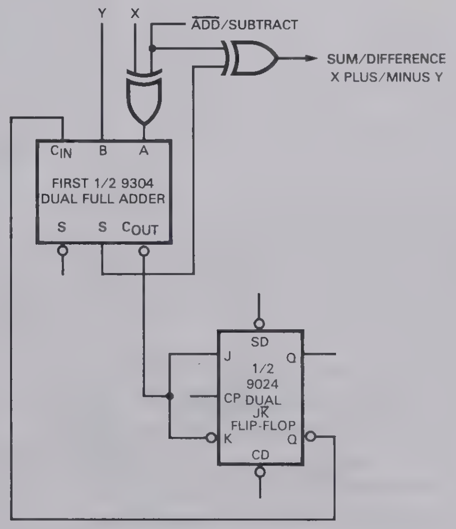
SERIAL BINARY ADDITION


B
A
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
S = A PLUS B
SD
J
Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
Q
CD

Half a 9304 dual full adder and one half of a 9024 dual flip-flop perform serial binary addition. For active High operands the carry flip-flop must be set when the least significant bit is applied. For active Low operands, the flip-flop must be reset when the least significant bit is applied.

SERIAL BINARY ADDITION/SUBTRACTION

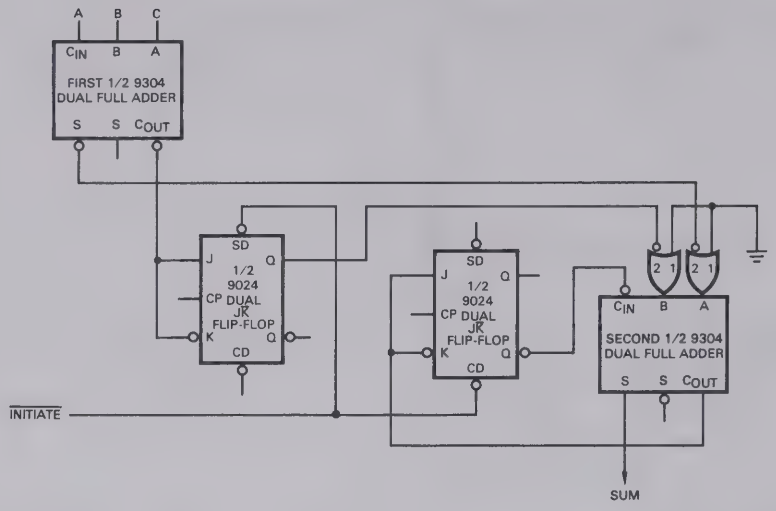

ADD/$\overline{\text{SUBTRACT}}$
H
B
A
2 1
2 1
CIN
B
A
SECOND 1/2 9304
DUAL FULL ADDER
S
S
COUT
S = A PLUS/MINUS B
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
SD
J
Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
Q
CD


Y
X
$\overline{\text{ADD}}$/SUBTRACT
SUM/DIFFERENCE
X PLUS/MINUS Y
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
SD
J
Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
Q
CD

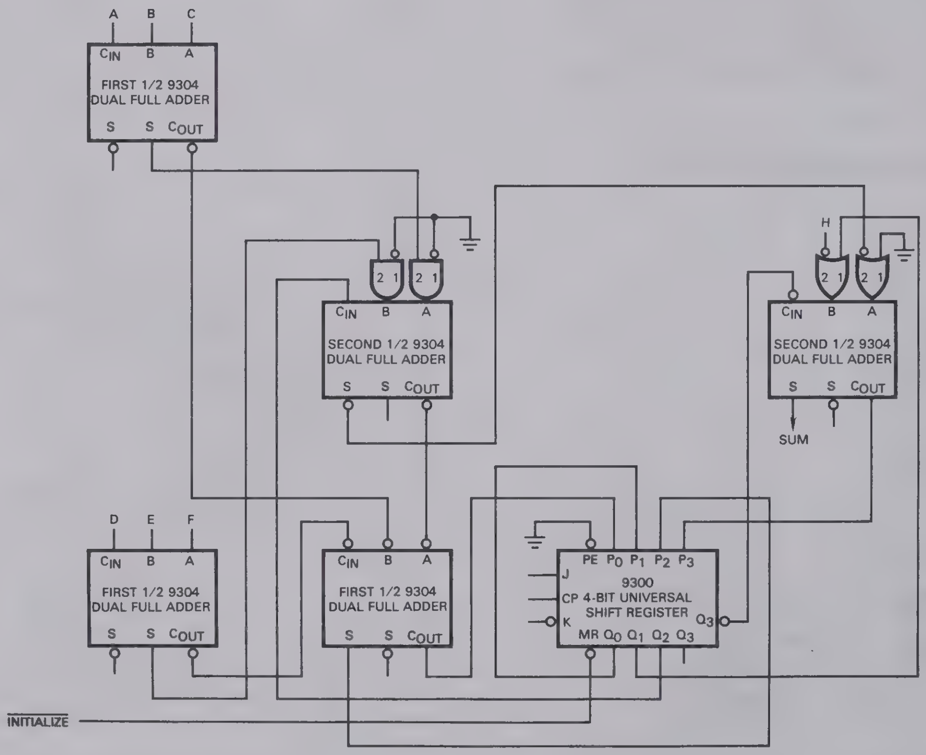
The most obvious design of a serial adder/subtractor inverts the B input for subtraction, using the other half of the 9304 as a conditional inverter. This design requires either a second pass for end around carry or it requires that the carry flip-flop starts out set for add, reset for subtract (with active High operands, opposite with active Low operands).

This second pass is avoided by using two Exclusive-OR gates (1/2 9014) in the data path, thereby effectively using the adder with active High operands in one mode, with active Low operands in the other. For both addition and subtraction, the carry flip-flop must start out set for active High operands, reset for active Low operands.

SERIAL ADDITION OF MORE THAN TWO VARIABLES


A
B
C
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
SD
J
Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
Q
CD
SD
J
Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
Q
CD
2 1
2 1
CIN
B
A
SECOND 1/2 9304
DUAL FULL ADDER
S
S
COUT
INITIATE
SUM

a

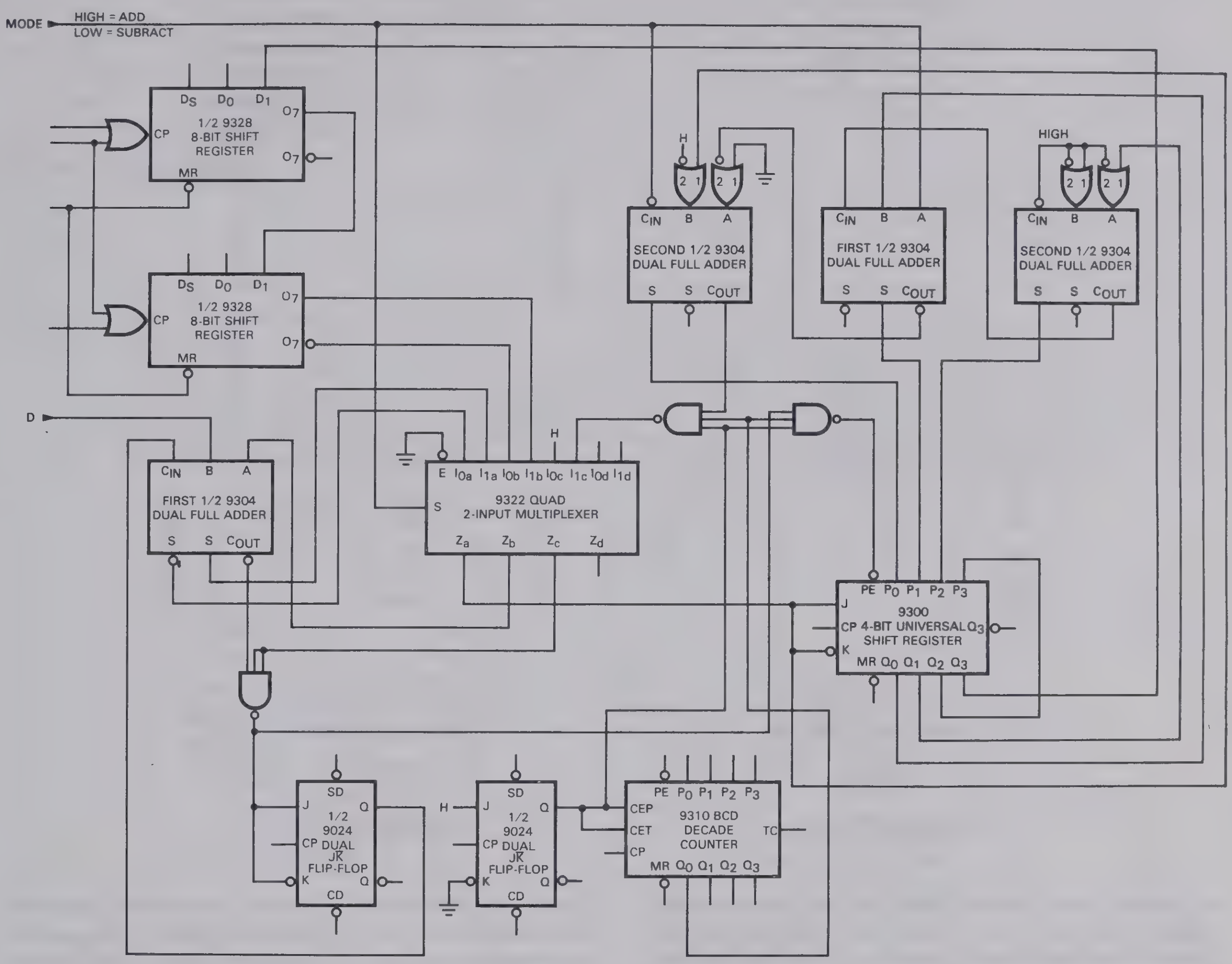

A
B
C
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
2 1
2 1
CIN
B
A
SECOND 1/2 9304
DUAL FULL ADDER
S
S
COUT
H
2 1
2 1
CIN
B
A
SECOND 1/2 9304
DUAL FULL ADDER
S
S
COUT
SUM
D
E
F
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
CIN
B
A
FIRST 1/2 9304
DUAL FULL ADDER
S
S
COUT
PE
P0
P1
P2
P3
J
9300
CP 4-BIT UNIVERSAL
SHIFT REGISTER
K
Q3
MR
Q0
Q1
Q2
Q3
INITIALIZE

b

The 9304 dual full adder can be used to add more than two operands. A 3-input adder (S_1 = A plus B plus C) using one 9304 and one 9024 dual flip-flop is shown in *a*.

A 6-input adder (S = A plus B plus C plus D plus E plus F) using 2 1/2 9304s and one 9300 4-bit register is shown in *b*. The adders are arranged such that all Carry outputs are active High, allowing the 9300 4-bit register with common asynchronous clear to be used for carry storage. This decreases package count, simplifies interconnections and decreases clock and clear loading.

5-DIGIT BIT-SERIAL BCD ADDER SUBTRACTOR (ACCUMULATOR)


MODE
HIGH = ADD
LOW = SUBRACT
DS D0 D1
1/2 9328
8-BIT SHIFT
REGISTER
CP
O7
O7
MR
DS D0 D1
1/2 9328
8-BIT SHIFT
REGISTER
CP
O7
O7
MR
D
CIN B A
FIRST 1/2 9304
DUAL FULL ADDER
S S COUT
H
E I0a I1a I0b I1b I0c I1c I0d I1d
S
9322 QUAD
2-INPUT MULTIPLEXER
Za Zb Zc Zd
H
2 1
2 1
CIN B A
SECOND 1/2 9304
DUAL FULL ADDER
S S COUT
CIN B A
FIRST 1/2 9304
DUAL FULL ADDER
S S COUT
HIGH
2 1
2 1
CIN B A
SECOND 1/2 9304
DUAL FULL ADDER
S S COUT
PE P0 P1 P2 P3
J
9300
CP 4-BIT UNIVERSAL Q3
SHIFT REGISTER
K
MR Q0 Q1 Q2 Q3
SD
J Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K Q
CD
H
SD
J Q
1/2
9024
CP DUAL
JK
FLIP-FLOP
K Q
CD
PE P0 P1 P2 P3
CEP
CET
CP
9310 BCD
DECADE
COUNTER
TC
MR Q0 Q1 Q2 Q3

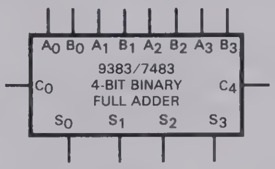
Bit serial operation of the 9304 is slower but requires fewer components than parallel addition/subtraction. A 5-digit bit serial BCD adder/subtractor makes use of the functional flexibility of the 9304. A 9328 16-bit shift register and a 9300 4-bit shift register form a 20-bit (5 digit) serial BCD accummulator. The modulo 20 counter (1/2 9024, 9310) keeps track of the data position in this register. When the counter is at count zero the LSB is stored in the last position of the 9328, the MSB in the first position of the 9300 (home position). Data can now be entered serially on the D input of the 9304 while the mode input defines the mode of operation (High for ADD, Low for SUBTRACT).

When adding, the first 9304 adder performs serial binary addition (active High Data, Carry inputs and Sum output, active Low Carry output). The Carry Out is stored in a carry flip-flop (1/2 9024). The other three full adders are used for BCD correction, adding six whenever the digit sum exceeds nine. The Carry output of this 3-bit adder is strobed whenever the most significant bit (weight 8) of each digit is at the data input and in the first adder. A carry out of either adder is shifted into the decimal carry flip-flop and if there is a decimal carry, the output of the 3-bit adder is parallel loaded into the 9300, adding six to the binary result (BCD correction).

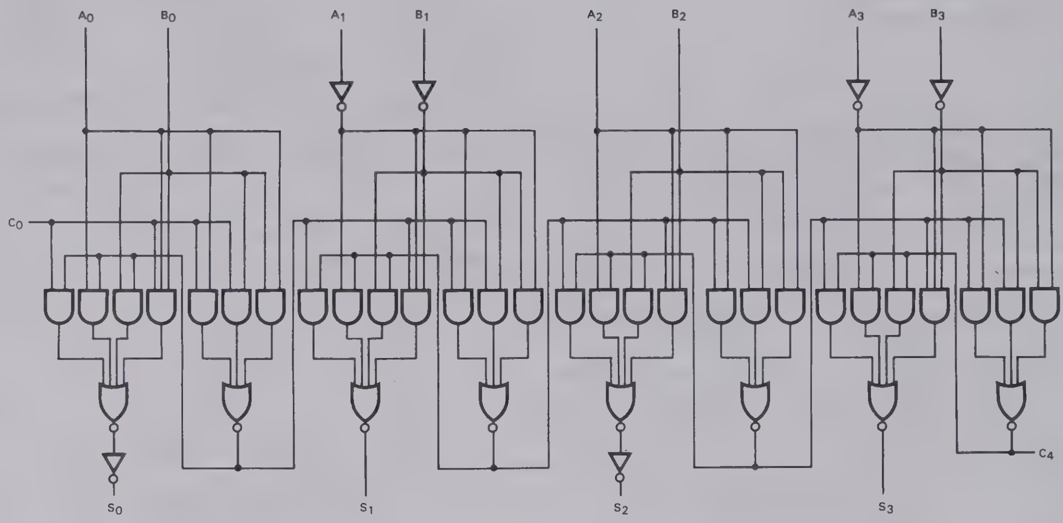
For subtraction the mode line must be Low and this mode line may only be changed when the counter is in the home position, possibly requiring a flip-flop. In this mode, the first adder has active Low Data and Carry In and an active High Carry Out. However, with respect to the shift registers, this is compensated by the change in the 9322 multiplexer.

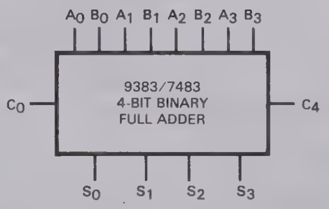
Because the carry flip-flop is reset operation starts with an automatic carry bit added to the LSB, required for 10 complement subtraction. The 3-bit adder adds 10 to each digit. A decimal borrow is indicated by a Low output from the first adder when it contains the MSB (weight 8) of each digit. This loads the adder output into the 9300, effectively adding 10 – subtracting six for BCD correction.

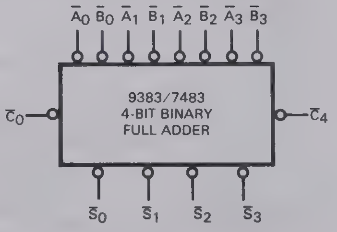
9383/7483 4-BIT RIPPLE CARRY ADDER


A0 B0 A1 B1 A2 B2 A3 B3
9383/7483
4-BIT BINARY
FULL ADDER
C0
C4
S0 S1 S2 S3

LEADS		LOADING
A_0, B_0, A_2, B_2	Data Inputs	4 UL
A_1, B_1, A_3, B_3	Data Inputs	1 UL
C_0	Carry Input	4 UL
S_0, S_1, S_2, S_3	Sum Outputs	10 UL
C_4	Carry Output	5 UL


A0
B0
A1
B1
A2
B2
A3
B3
C0
C4
S0
S1
S2
S3

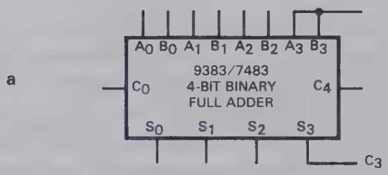

A0 B0 A1 B1 A2 B2 A3 B3
9383/7483
4-BIT BINARY
FULL ADDER
C0
C4
S0 S1 S2 S3

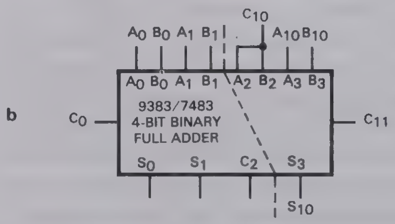

Ā0 B̄0 Ā1 B̄1 Ā2 B̄2 Ā3 B̄3
9383/7483
4-BIT BINARY
FULL ADDER
C̄0
C̄4
S̄0 S̄1 S̄2 S̄3

As with all other adders, 9383 operation can be described in terms of both active High and active Low inputs and outputs, (refer to dual logic representations). Note that with active High inputs, Carry In can not be left open, but must be held Low when no carry-in is intended.

The 9383/7483 4-bit ripple carry adder logic diagram and loading rules are shown with inputs and outputs numbered from 0 to 3, consistent with 9300 family terminology. This device provides 4-bit addition in a single 16-lead package, very useful in applications which do not require the speed and functional versatility of more sophisticated 4-bit Arithmetic Logic Units.

THE 9383 AS A 3-BIT OR A 2 + 1-BIT ADDER

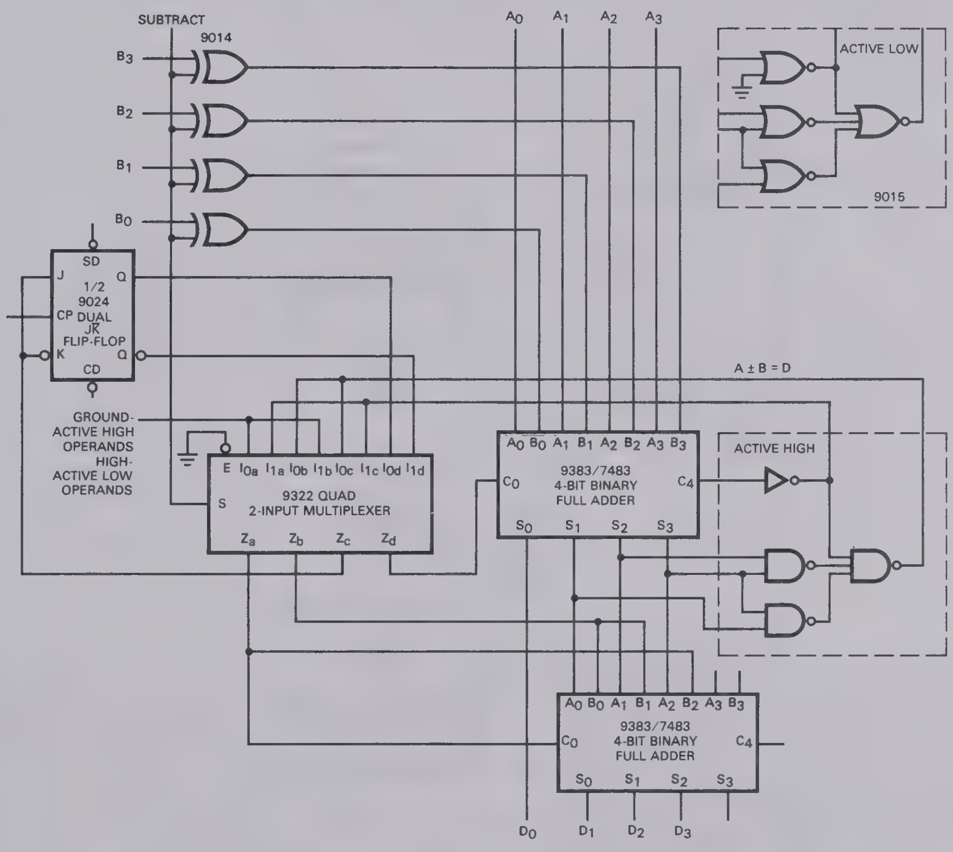

a
A0 B0 A1 B1 A2 B2 A3 B3
9383/7483
4-BIT BINARY
FULL ADDER
C0
C4
S0 S1 S2 S3
C3


b
C10
A0 B0 A1 B1
A10 B10
A0 B0 A1 B1 A2 B2 A3 B3
9383/7483
4-BIT BINARY
FULL ADDER
C0
C11
S0 S1 C2 S3
S10

Because all 16 leads of the 9383 are needed for basic input, output, and supply connections, functional versatility is minimal. The 9383 can be connected as a 3-bit adder *(a)* with carry input and output by using the least significant data inputs and the Carry In; normally tying A_3 and B_3 together and terminating them either High or Low. The Carry Out signal is available on the S_3 output.

The 9383 is useful as two independent adders, *(b)* one two bits wide, the other one bit wide. The two least significant bits are used as a 2-bit adder with Carry Out on the S_2 output. Inputs A_2 and B_2 are tied together and used as carry input for the second adder. The A and B operands of the second adder applied on the A_3 and B_3 inputs, carry-out appears on the C_4 output. All these configurations work with active High as well as active Low inputs and outputs.

BCD ADDER/SUBTRACTOR

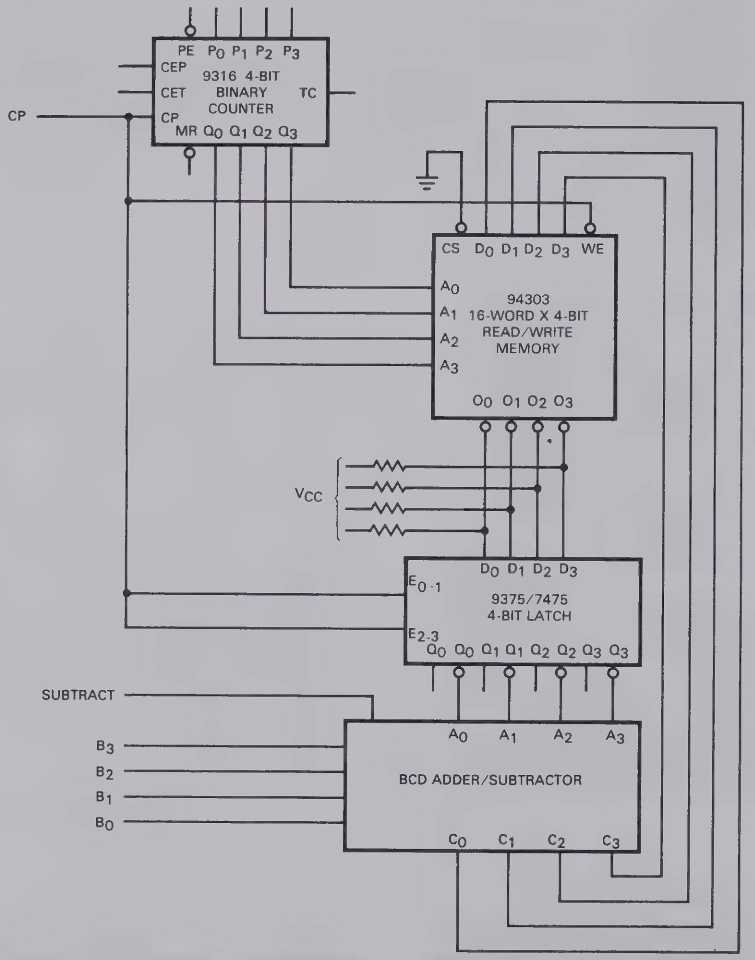

SUBTRACT
9014
B_3
B_2
B_1
B_0
A_0
A_1
A_2
A_3
ACTIVE LOW
9015
SD
J
Q
1/2
9024
CP DUAL
$\overline{JK}$
FLIP-FLOP
K
$\overline{Q}$
CD
A ± B = D
GROUND-
ACTIVE HIGH
OPERANDS
HIGH-
ACTIVE LOW
OPERANDS
E I_{0a} I_{1a} I_{0b} I_{1b} I_{0c} I_{1c} I_{0d} I_{1d}
S
9322 QUAD
2-INPUT MULTIPLEXER
Z_a Z_b Z_c Z_d
A_0 $\overline{B_0}$ A_1 B_1 A_2 B_2 A_3 B_3
C_0
9383/7483
4-BIT BINARY
FULL ADDER
C_4
S_0 S_1 S_2 S_3
ACTIVE HIGH
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3
C_0
9383/7483
4-BIT BINARY
FULL ADDER
C_4
S_0 S_1 S_2 S_3
D_0
D_1
D_2
D_3

This circuit performs BCD corrected addition and subtraction on four bits (one digit) in parallel. For addition, the control input (SUBTRACT) is LOW, the first 9383 4-bit ripple-carry adders adds the B_{0-3} inputs the A_{0-3} inputs, generating the binary sum on outputs S_{0-3} and the binary carry on output C_4. Whenever the binary sum exceeds 9, i.e., when $S_3 \cdot (S_2 + S_1) + C_4$, a decimal carry is generated by the gating structure shown, setting the carry flip-flop and forcing a binary 6 onto the B inputs of the second 9383 4-bit adder. The outputs D_0–D_3 represent the BCD corrected sum D = A plus B.

For subtraction, the control input (SUBTRACT) is HIGH, inverting B_{0-3} inputs to the first 9383 adder. The 9322 multiplexer feeds the $\overline{Q}$ output of the carry flip-flop into the Carry In of the first 9383 which performs Carry plus A plus $\overline{B}$, the well-known algorithm for binary subtraction. The Carry Out (C_4) signal is inverted before it is routed through the multiplexer into the J • K input of the Carry Borrow flip-flop. Whenever this flip-flop is being set, the binary result at S_{0-3} requires correction, by subtracting 6 or adding 10. This is performed in the second 9383 by routing the $\overline{C}_4$ signal into the C_0 (weight 2) and the B_2 input (weight 8).

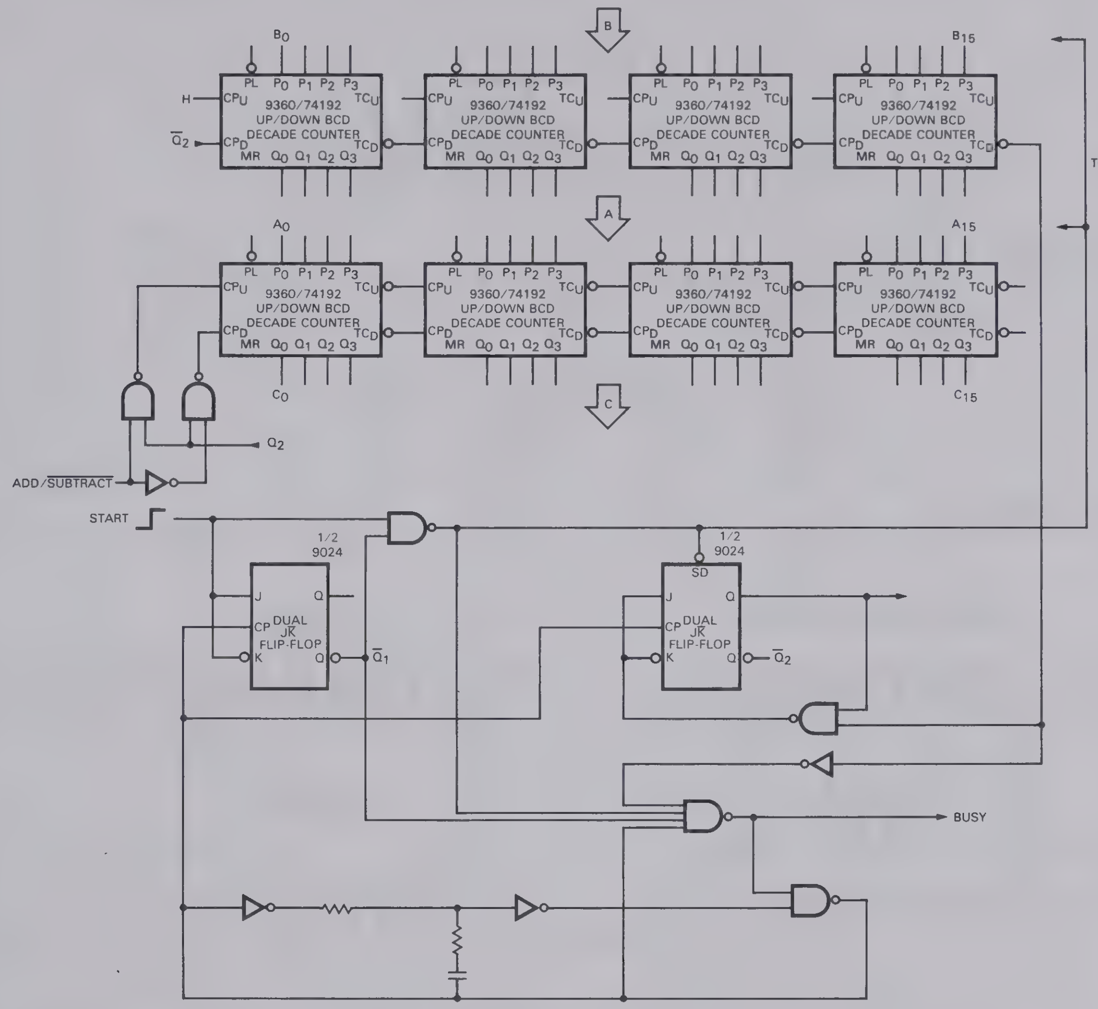
The outputs D_{0-3} represent the BCD corrected result D = A minus B. Since BCD addition is an asymmetrical function, the circuit must be slightly modified for active Low operands.

16-DIGIT BCD ACCUMULATOR


PE
P0 P1 P2 P3
CEP
9316 4-BIT BINARY COUNTER
CET
TC
CP
MR Q0 Q1 Q2 Q3
CP
CS D0 D1 D2 D3 WE
A0
A1
A2
A3
94303 16-WORD X 4-BIT READ/WRITE MEMORY
O0 O1 O2 O3
VCC
D0 D1 D2 D3
E0-1
9375/7475 4-BIT LATCH
E2-3
Q0 Q0 Q1 Q1 Q2 Q2 Q3 Q3
SUBTRACT
A0 A1 A2 A3
B3
B2
B1
B0
BCD ADDER/SUBTRACTOR
C0 C1 C2 C3

The same BCD adder/subtractor described in the preceding diagram can be used in a 16-digit BCD accumulator. The 9316 modulo 16 counter and the 93403, 16 word by 4-bit read/write memory form a 16-digit serial memory (serial by digit, parallel by bit). The 9316 increments and thus changes memory addresses on the rising edge of the clock. When the clock is High, the memory is in the READ mode ($\overline{WE}$ = High) and presents its contents on the active Low outputs O_{0-3}. When the clock is High, the 9375 4-bit latch with complement outputs is enabled, effectively re-inverting the memory outputs, cancelling the memory inversion. The BCD adder/subtractor adds the B inputs to, or subtracts them from, the accumulator. When the clock goes Low the 9375 latches are disabled, freezing the accumulator outputs, and preventing a race condition when the Low clock ($\overline{WE}$ = Low) enters the output of the adder/subtractor into the memory.

SLOW BCD ADDER/SUBTRACTOR/ ACCUMULATOR USING COUNTERS



For very slow applications, BCD addition/subtraction can be economically performed with counters. This circuit adds/subtracts (or accumulates) 4-digit BCD numbers in less than 3 ms, using eight up/down BCD counters, three gate packages, and a dual flip-flop for control. This uses considerably less hardware than an equivalent bit-serial or character-serial BCD adder/subtractor requires.

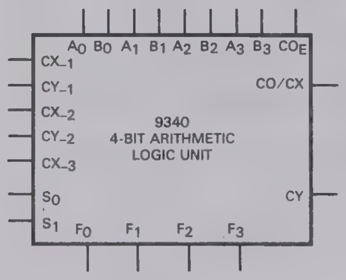
In the idle condition the Start input is Low, the Busy output is Low, Q_1, is Low, and the internal clock is High and steady. A Low-to-High transition of the Start input generates a parallel load signal to counters A and B, asynchronously presets Q_2, and starts generating a Busy output while starting the internal oscillator (≈6 MHz). During the first Low clock period, both counters are asynchronously loaded and Q_2 is asynchronously preset. The second Low-to-High clock transition resets Q_2 which generates an active Low clock pulse on the CP_D input of counter B and on either the CP_U (for addition), or the CP_D (for subtraction) input of counter A. The next Low-to-High clock transition terminates this pulse and causes both counters to change, provided the TC_D output of counter B is still High. This clocks the two counters together until the TC_D output of counter B goes Low indicating that the *previous* clock pulse decremented B to zero. The clock pulse that has already started must, therefore, neither increment or decrement counter A. This is accomplished by keeping the Q_2 High and leaving the clock to the counters Low. Thus, counter B is at zero and counter A contains the desired sum or difference.

If the Start input has gone Low, the output Busy line goes Low when the TC_D output from counter B goes Low. This causes the internal clock generator to stop indicating the operation is finished. If the Start input has not gone Low, output Busy stays High. The internal clock continues, but both counters remain fixed. The first clock pulse after Start goes Low terminates output Busy, stops the clock, and ends the operation.

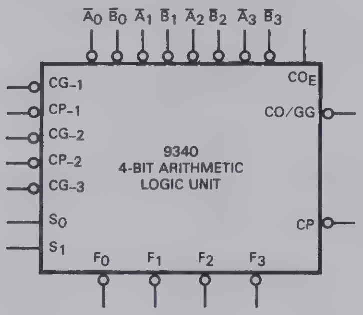
This circuit is easily converted to an accumulator by keeping the PL inputs of the A counter High. Such a circuit operates in either binary or BCD notation, as long as the appropriate counters are used (9366 for binary, 9360 for BCD). Binary operation offers no advantage because a straightforward ripple carry adder using 9383s for both faster and more economical. Using binary devices in one counter and BCD devices in the other makes this circuit a code converter (binary-to-BCD or BCD-to-binary).

9340 ARITHMETIC LOGIC UNIT

ACTIVE HIGH (ALTERNATE)


A0 B0 A1 B1 A2 B2 A3 B3 COE
CX-1
CY-1
CX-2
CY-2
CX-3
S0
S1
F0 F1 F2 F3
CO/CX
CY
9340
4-BIT ARITHMETIC
LOGIC UNIT

ACTIVE LOW


Ā0 B̄0 Ā1 B̄1 Ā2 B̄2 Ā3 B̄3
CG-1
CP-1
CG-2
CP-2
CG-3
S0
S1
F0 F1 F2 F3
COE
CO/GG
CP
9340
4-BIT ARITHMETIC
LOGIC UNIT

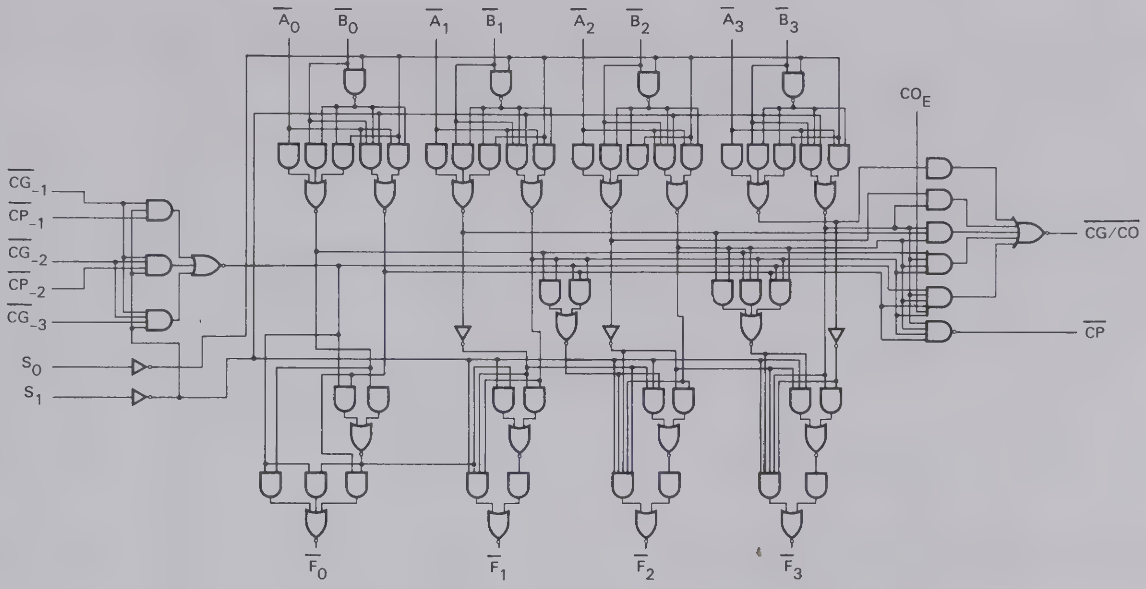
LEADS		LOADING (*)
$\overline{A}_0$ to $\overline{A}_3$,$\overline{B}_0$ to $\overline{B}_3$	Operand Active Low Inputs	3 UL
S_0,S_1	Mode Select Inputs	1 UL
$\overline{CG}_1$	Active Low Carry Generate Input from immediately preceding stage	3 UL
$\overline{CP}_1$	Active Low Carry Propagate Input from immediately preceding stage	1 UL
$\overline{CG}_2$	Active Low Carry Generate Input from second preceding stage	2 UL
$\overline{CP}_2$	Active Low Carry Propagate Input from second preceding stage	1 UL
$\overline{CG}_3$	Active Low Carry Generate Input from third preceding stage	1 UL
COE	Carry Out Enable Input	1.5 UL
$\overline{F}_0$,$\overline{F}_1$,$\overline{F}_2$,$\overline{F}_3$	Function (Active Low) Outputs (**)	10 UL
$\overline{CO/CG}$	Carry Out/Carry Generate (Active Low) Output (**)	10 UL
$\overline{CP}$	Carry Propagate (Active Low) Output (**)	10 UL

*1 Unit Load (UL) = 40µA High/1.6 mA Low
**10 UL is the output Low drive factor and 20 UL is the output High drive factor.

The 9340 MSI Arithmetic Logic Unit is a high speed combinatorial circuit capable of performing not only addition and subtraction, but also several logic functions on two 4-bit binary words. Internally, the device uses carry lookahead logic for its high speed. Provision is made for carry lookahead interconnections between several 9340s without the need for additional logic.

The ALU is suitable for use in general and special purpose digital computers as the center of high speed arithmetic units. The 9340 can perform arithmetic on binary numbers in ones complement or twos complement. The input data can be either active High or active Low.

9340 OPERATION



The 9340, functionally, can be divided into several parts. At the top is a set of gates that produce the AND and OR functions of the A and B inputs.

$$A_iB_i = \&_i$$

$$A_i + B_i = V_i$$

The gates that form the & and V functions are under the control of the S_0 and S_1 inputs so that several different AND and OR functions can be formed. The functions, instead of being AB and A + B, may be $A\overline{B}$ and $A + \overline{B}$. This control is used to vary the functions performed by the circuit. The two sets of functions cited above are typically used for addition and subtraction, respectively. At the bottom of the figure shown is a set of adders, which add (Ex OR) the & and V signals for a particular bit, and the appropriate carry in. At the left of the logic diagram is a set of gates producing a carry in function. These gates accept $\overline{CG}$ and $\overline{CP}$ outputs from other 9340s to produce a carry in. Carry In =

$$\overline{\overline{CG}_{-1} \cdot \overline{CP}_{-1} + \overline{CG}_{-1} \cdot \overline{CG}_{-2} \cdot \overline{CP}_{-2} + \overline{CG}_{-1} \cdot \overline{CG}_{-2} \cdot \overline{CG}_{-3}}$$

The subscripts –1, –2, and –3 refer to the preceding devices. A carry in to one 9340 is produced by a Carry Generate from the immediately preceding device (CG_{-1}) or by a propagate from the preceding device and a Carry Generate from a 9340 two devices back (CP_{-1} CG_{-2}), etc. One of the select lines, S_1, blocks the propagation of carries between bits. This control is used to select logic or arithmetic functions. For example, subtraction is basically $A \oplus \overline{B} \oplus$ Carry.

If S_1 is High, carries are forced at all bits and the output becomes $A \oplus \overline{B} \oplus 1 = A \oplus B$.

At the right of the figure is a series of gates producing look ahead carry out functions. One gate produces an active Low carry propagate over the block of four bits. Logically, it is the AND of the four internal Carry Propagates. Functionally, it means that if there is a carry-in to this package, there should be a carry out to the next package. It is activated by the presence of 1111 in the 9340. $CP = V_3V_2V_1V_0$

	MSB			LSB	
Word A	1	0	1	0	
Word B	1	1	0	1	
Sum	1	1	1	1	"Carry Propagate"

The other gate forms an active Low Carry Generate for the block of four bits. The CG is a carry out from this particular device.

$$CG = \&_3 + V_3\&_2 + V_3V_2\&_1 + V_3V_2V_1\&_0$$

Note that neither CG nor CP outputs are in any way affected by a carry in signal. They are only functions of the A and B operand inputs.

The CG signal can be turned into a true carry out if one additional term is added to include a carry in condition. This term is controlled by the COE lead (Carry Out Enable). Since Carry Out = $CG + CP\ \ C_{IN}$, for the 9340 $CG/CO = CG + CP\ C_{IN}\ COE$. The internal carry signals for the 9340 are defined by the following equations ($C_0\ C_1\ C_2$ refer to the internal carry signals to the four bits):

$$C_{IN} = CG_{-1} + CP_{-1}\ CG_{-2} + CP_{-1}\ CP_{-2}\ CG_{-3}$$

$$C_0 = S_1 + C_{IN}$$

$$C_1 = S_1 + (\&_0 + V_0\ C_{IN})$$

$$C_2 = S_1 + (\&_1 + V_1\ \&_0 + V_1\ V_0\ C_{IN})$$

$$C_3 = S_1 + (\&_2 + V_2\ \&_1 + V_2\ V_1\ \&_0 + V_2\ V_1\ V_0\ C_{IN})$$

$$CP\ (output) = V_3\ V_2\ V_1\ V_0$$

$$CG\ (output) = (\&_3 + V_3\ \&_2 + V_3\ V_2\ \&_1 + V_3\ V_2\ V_1\ \&_0 + V_3\ V_2\ V_1\ V_0\ C_{IN}\ COE)$$

9340 FUNCTIONS FOR ACTIVE LOW A, B, AND F

The S_0 and S_1 inputs control the function of the 9340 in two ways. First, they determine the particular AND and OR functions produced by the input gating, and second, the S_1 input determines whether or not carries are propagated between the bits in the 9340. The functions of the 9340 are outlined in the table. Each F_i output is $\overline{\&}_i V_i \oplus \text{Carry}_i$. If S_1 is High, a carry is forced and the function becomes $\overline{\&}_i V_i \oplus 1$. The table is for active Low A, B, and F.

S_0	S_1	&	V	$\overline{\&}V \oplus \text{Carry}$	Function
L	L	$A\overline{B}$	$A+\overline{B}$	$A \oplus \overline{B} \oplus \text{Carry}$	A subtract B
H	L	AB	$A+B$	$A \oplus B \oplus \text{Carry}$	A add B
L	H	$A\overline{B}$	$A+\overline{B}$	$A \oplus \overline{B} \oplus 1$	A Ex-OR B
H	H	AB	1	$\overline{AB} \oplus 1$	A AND B

9340 FUNCTIONS FOR ACTIVE HIGH B AND ACTIVE LOW A AND F

The corresponding function for inputs and outputs of various polarities may be determined by making the required inversion of the variables. For example, if B is active High and A and F are active Low, the functions are found by replacing B with $\overline{B}$.

S_0	S_1	$\overline{\&}V \oplus \text{Carry}$	Function
L	L	$A \oplus \overline{\overline{B}} \oplus \text{Carry}$	A add B
H	L	$A \oplus \overline{B} \oplus \text{Carry}$	A subtract B
L	H	$A \oplus \overline{\overline{B}} \oplus 1$	A compare B
H	H	$\overline{A\overline{B}} \oplus 1$	A AND $\overline{B}$

9340 FUNCTIONS FOR ACTIVE HIGH A, B, AND F

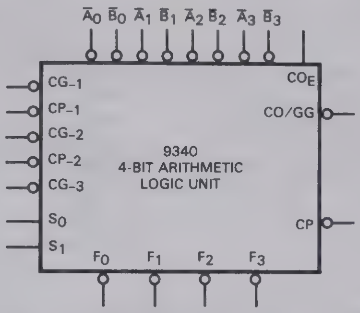
The functions for active High A and B and active High F are found by replacing A and B and carry in the first table with $\overline{A}$ and $\overline{B}$ and $\overline{\text{carry}}$ and then inverting the entire function.

S_0	S_1	$\overline{\&}V \oplus \text{Carry}$	Function
L	L	$\overline{A} \oplus \overline{\overline{B}} \oplus \overline{\text{Carry}} = \overline{A \oplus \overline{B} \oplus \text{Carry}}$	A subtract B
H	L	$\overline{A} \oplus \overline{B} \oplus \overline{\text{Carry}} = \overline{A \oplus B \oplus \text{Carry}}$	A add B
L	H	$\overline{A} \oplus \overline{\overline{B}} \oplus 1 = \overline{\overline{A \oplus \overline{B}}}$	A compare B
H	H	$\overline{\overline{A}\,\overline{B}} \oplus 1 = \overline{A + B}$	A OR B

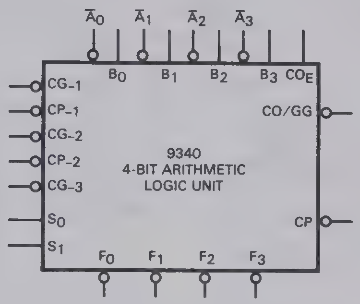
ADDERS/SUBTRACTORS

9340 LOGIC EQUIVALENTS

ACTIVE LOW

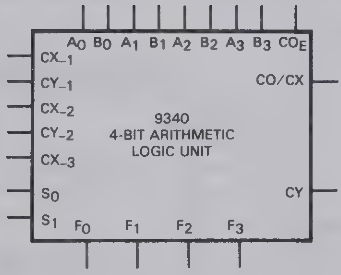

$\overline{A}_0$ $\overline{B}_0$ $\overline{A}_1$ $\overline{B}_1$ $\overline{A}_2$ $\overline{B}_2$ $\overline{A}_3$ $\overline{B}_3$
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
S_0
S_1
CO_E
CO/GG
CP
9340
4-BIT ARITHMETIC
LOGIC UNIT
F_0 F_1 F_2 F_3

S_0	S_1	Function
L	L	A subtract B
H	L	A add B
L	H	A EX OR B
H	H	A AND B

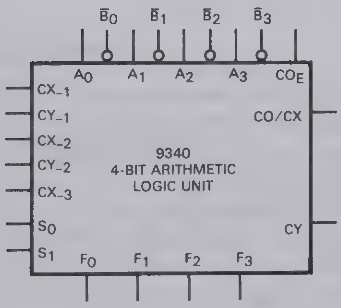

$\overline{A}_0$ $\overline{A}_1$ $\overline{A}_2$ $\overline{A}_3$
B_0 B_1 B_2 B_3 CO_E
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
S_0
S_1
CO/GG
CP
9340
4-BIT ARITHMETIC
LOGIC UNIT
F_0 F_1 F_2 F_3

S_0	S_1	Function
L	L	A add B
H	L	A subtract B
L	H	A equivalent B
H	H	A AND $\overline{B}$

ACTIVE HIGH


A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 CO_E
CX_{-1}
CY_{-1}
CX_{-2}
CY_{-2}
CX_{-3}
S_0
S_1
CO/CX
CY
9340
4-BIT ARITHMETIC
LOGIC UNIT
F_0 F_1 F_2 F_3

S_0	S_1	Function
L	L	A subtract B
H	L	A add B
L	H	A equivalent B
H	H	A OR B


$\overline{B}_0$ $\overline{B}_1$ $\overline{B}_2$ $\overline{B}_3$
A_0 A_1 A_2 A_3 CO_E
CX_{-1}
CY_{-1}
CX_{-2}
CY_{-2}
CX_{-3}
S_0
S_1
CO/CX
CY
9340
4-BIT ARITHMETIC
LOGIC UNIT
F_0 F_1 F_2 F_3

S_0	S_1	Function
L	L	A add B
H	L	A subtract B
L	H	A EX OR B
H	H	A OR $\overline{B}$

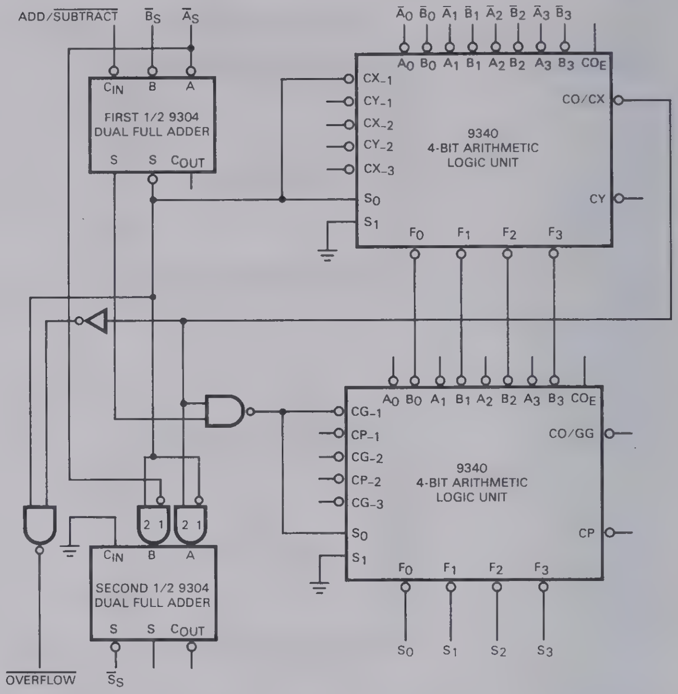
Functions for all useful representations of the 9340 are shown in Logic Equivalents. The output signals labeled CG and CP are "carry generate" and "carry propagate" only in the active Low representation. When the operands are active High, the nature of the CG and CP signals changes. The CG output becomes similar to an active High CP, and the CP output becomes similar to an active High CG. It is important to recognize at this point that an adder is always an adder. Once the carry signals are connected, the polarities of the inputs and outputs of an adder can be changed, but the device is still an adder.

This implies that even though the carry signals from the 9340 are no longer "carry propagate" and "carry generate", they can be connected to other devices exactly as if they were, and the adder will operate properly. To be accurate, the carry signals have been relabeled CX and CY in the active High case.

ADDERS/SUBTRACTORS

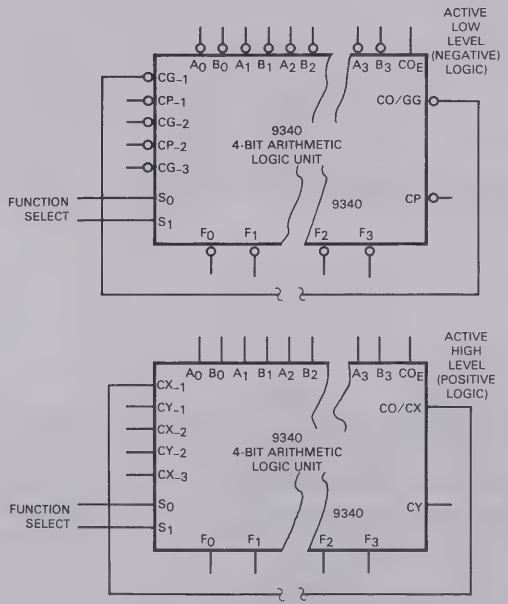
SIGN MAGNITUDE ADDITION AND SUBTRACTION

The 9340 can be used to perform arithmetic operations in all three binary signed number representations, including the most complex, sign magnitude. Sign magnitude is complex because when the signs are different — and an addition takes place, or when the signs are the same and a subtraction takes place — the absolute *magnitude* of the difference between the numbers is required. There is also the problem that the sign bit must be handled separately. The figure shows a sign magnitude 5-bit Arithmethic Logic Unit that uses two 9340s, one for performing addition and subtraction on the two operands and the other to form the twos complement of the result, if the result of the previous operation is negative. The scheme can be extended for larger word lengths; unused inputs at the most significant end of the first level of ALUs should have the $\overline{A}$ inputs at ground and the $\overline{B}$ inputs at a High logic level. Because of these complications, sign magnitude arithmetic is rarely used except where a small amount of processing is involved and the operands are available in the sign magnitude representation. In most systems it is more efficient to work in complement arithmetic converting to sign magnitude as required.



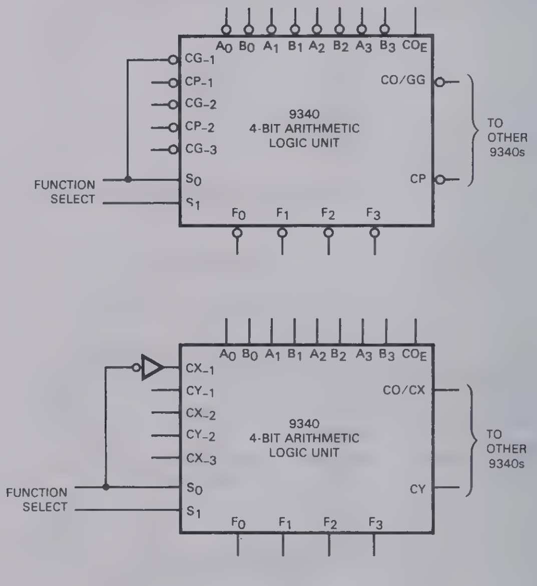
ADD/$\overline{SUBTRACT}$
$\overline{B}_S$
$\overline{A}_S$
C_{IN} B A
FIRST 1/2 9304 DUAL FULL ADDER
S S C_{OUT}
$\overline{A}_0$ $\overline{B}_0$ $\overline{A}_1$ $\overline{B}_1$ $\overline{A}_2$ $\overline{B}_2$ $\overline{A}_3$ $\overline{B}_3$
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 CO_E
CX_{-1}
CY_{-1}
CX_{-2}
CY_{-2}
CX_{-3}
CO/CX
9340 4-BIT ARITHMETIC LOGIC UNIT
S_0
CY
S_1
F_0 F_1 F_2 F_3
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
CO/GG
CP
2 1
2 1
C_{IN} B A
SECOND 1/2 9304 DUAL FULL ADDER
S S C_{OUT}
$\overline{OVERFLOW}$
$\overline{S}_S$
S_0 S_1 S_2 S_3

COMPLEMENT ARITHMETIC



ACTIVE LOW LEVEL (NEGATIVE) LOGIC)
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 CO_E
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
CO/GG
9340 4-BIT ARITHMETIC LOGIC UNIT
FUNCTION SELECT
S_0
S_1
9340
CP
F_0 F_1 F_2 F_3
ACTIVE HIGH LEVEL (POSITIVE LOGIC)
CX_{-1}
CY_{-1}
CX_{-2}
CY_{-2}
CX_{-3}
CO/CX
CY

ONES COMPLEMENT



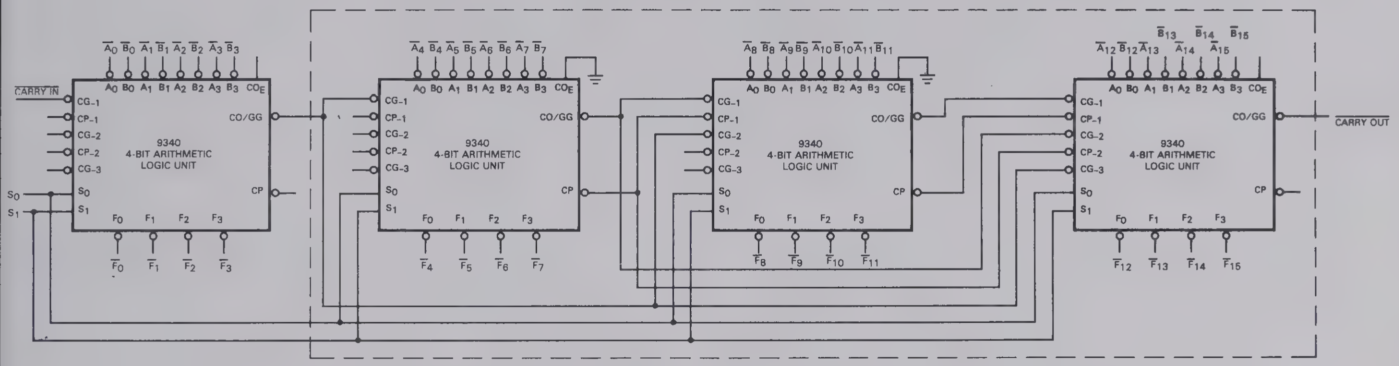
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 CO_E
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
CO/GG
9340 4-BIT ARITHMETIC LOGIC UNIT
TO OTHER 9340s
FUNCTION SELECT
S_0
S_1
CP
F_0 F_1 F_2 F_3
CX_{-1}
CY_{-1}
CX_{-2}
CY_{-2}
CX_{-3}
CO/CX
CY

TWOS COMPLEMENT

Some machines use ones complement arithmetic because of the apparent simplicity of performing subtraction in this representation. The main disadvantage is that the end around carry causes extra delay, and only three ALUs can be cascaded in the first section of systems using 9340s. The connections are shown.

The most popular method used in binary machines is twos complement arithmetic. The sign bit is treated in an identical manner to the rest of the word and, instead of an end around carry, a carry in is immediately forced when a subtraction takes place. This can conveniently be done in an ALU built using 9340s by connecting the S_0 input to the $\overline{CG}_{-1}$ input of the first ALU in the chain. All applications performing arithmetic assume that the twos complement representation is used. The 16-bit Arithmetic Logic Unit shown is for numbers represented in the two complement notation using active Low operands; for active High operands the connection is the same except that an inverter must be inserted between the S_0 input and the CX_{-1} line.

16-BIT FULL LOOKAHEAD SYSTEM



$\overline{A_0}$ $\overline{B_0}$ $\overline{A_1}$ $\overline{B_1}$ $\overline{A_2}$ $\overline{B_2}$ $\overline{A_3}$ $\overline{B_3}$
$\overline{A_4}$ $\overline{B_4}$ $\overline{A_5}$ $\overline{B_5}$ $\overline{A_6}$ $\overline{B_6}$ $\overline{A_7}$ $\overline{B_7}$
$\overline{A_8}$ $\overline{B_8}$ $\overline{A_9}$ $\overline{B_9}$ $\overline{A_{10}}$ $\overline{B_{10}}$ $\overline{A_{11}}$ $\overline{B_{11}}$
$\overline{A_{12}}$ $\overline{B_{12}}$ $\overline{A_{13}}$ $\overline{B_{13}}$ $\overline{A_{14}}$ $\overline{B_{14}}$ $\overline{A_{15}}$ $\overline{B_{15}}$
$\overline{\text{CARRY IN}}$
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 CO_E
CG_{-1}
CP_{-1}
CG_{-2}
CP_{-2}
CG_{-3}
CO/GG
CP
S_0
S_1
F_0 F_1 F_2 F_3
9340
4-BIT ARITHMETIC
LOGIC UNIT
$\overline{F_0}$ $\overline{F_1}$ $\overline{F_2}$ $\overline{F_3}$
$\overline{F_4}$ $\overline{F_5}$ $\overline{F_6}$ $\overline{F_7}$
$\overline{F_8}$ $\overline{F_9}$ $\overline{F_{10}}$ $\overline{F_{11}}$
$\overline{F_{12}}$ $\overline{F_{13}}$ $\overline{F_{14}}$ $\overline{F_{15}}$
$\overline{\text{CARRY OUT}}$

TYPICAL DELAY TABLE

WORD LENGTH (bits)	ADD (ns)	SUBTRACT (ns)
1-4	23	27
5-16	38	41
17-28	53	56
19-40	68	71
41-52	83	86
53-64	98	101
65-76	113	116
77-88	128	131
89-100	143	146

Additional blocks of 12 ALU stages can be connected to form ripple block addition and give very high speed arithmetic over large word lengths with no extra carry circuitry required. Since only two additional gate delays are incurred for each 12-bit increment, and when a separate carry lookahead package is used, two delays are incurred for each level of lookahead, the 9340 is at least as fast as a 2-level lookahead for up to 28-bit words.

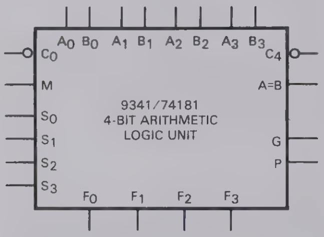
Four 9340 ALUs can be connected to form a 16-bit full lookahead ALU. This ALU can work in one or two complement arithmetic representations and in the active Low or active High logic representations. If longer word lengths are required, 12-bit ALU blocks connected as shown in the dotted area of the diagram can be cascaded at the end of the 16-bit full lookahead portion.

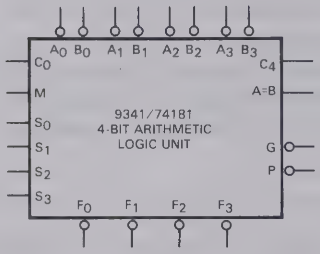
COMPARISON FUNCTIONS

Magnitude comparison of the inputs to the 9340 is performed with the device in the subtract mode and observing the Carry Out signal. In a twos complement subtraction (with Carry In active), the function performed is A – B. If the Carry Out signal from the sign is active, then A is greater than or equal to B. If the Carry input is removed, the function is A minus B minus 1. If the Carry output is active for this operation, then A is greater than B. When only the first case is true, obviously A = B.

Equivalence between A and B may be detected directly by using the Carry Generate and Carry Propagate outputs of the 9340. If A and B are equal, then in the subtract mode or in the Exclusive OR mode, all the internal V signals will be active and none of the internal & signals will be active. This is detected at the outputs by the condition, Equivalence = $CP \bullet \overline{CG}$. Equivalence over the entire word is detected by forming the above function for each 9340 and ANDing all the signals.

9341 ARITHMETIC LOGIC UNIT

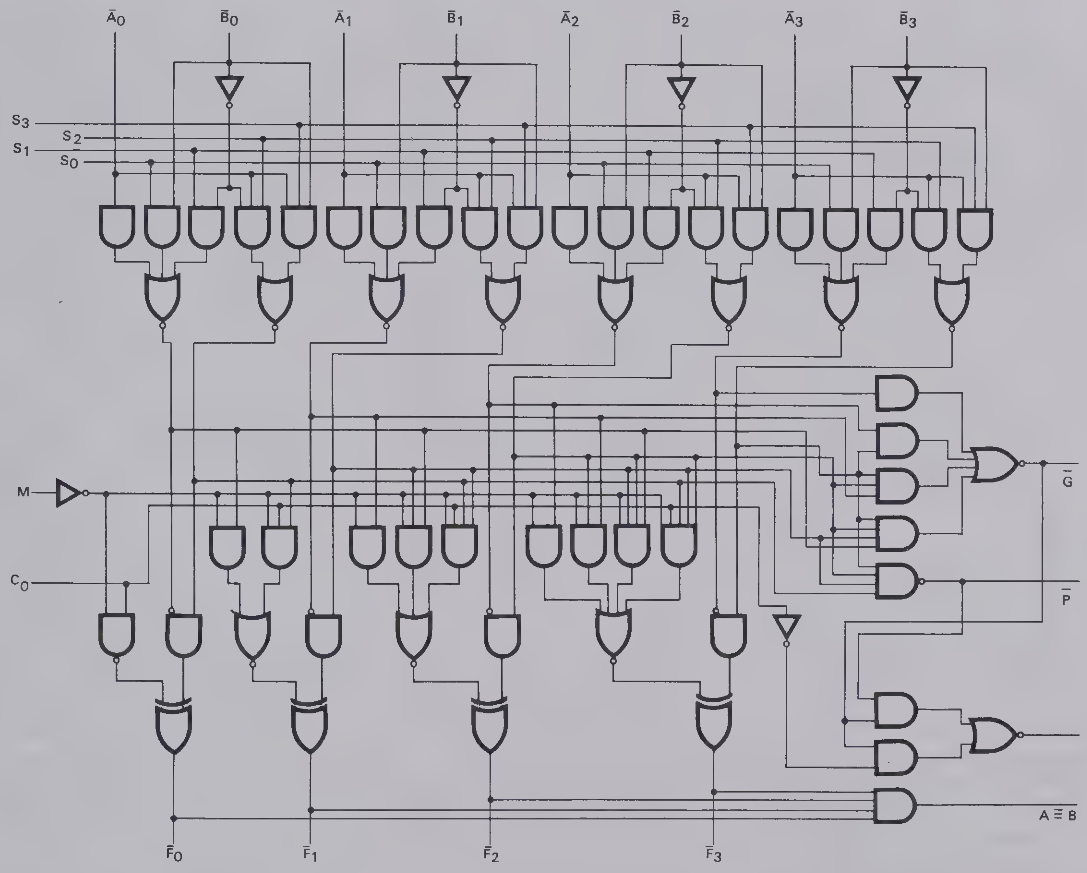

A0 B0 A1 B1 A2 B2 A3 B3
C0
C4
M
A=B
S0
S1
S2
S3
9341/74181
4-BIT ARITHMETIC
LOGIC UNIT
G
P
F0 F1 F2 F3


A0 B0 A1 B1 A2 B2 A3 B3
C0
C4
M
A=B
S0
S1
S2
S3
9341/74181
4-BIT ARITHMETIC
LOGIC UNIT
G
P
F0 F1 F2 F3

LEADS		LOADING
$\overline{A}_0 - \overline{A}_3, \overline{B}_0 - \overline{B}_3$	Operand (Active Low) Inputs	3 UL
S_0, S_1, S_2, S_3	Function — Select Inputs	4 UL
M	Mode Control Input	1 UL
C_0	Carry Input	5 UL
$\overline{F}_0, \overline{F}_1, \overline{F}_2, \overline{F}_3$	Function (Active Low) Outputs	10 UL
A=B	Comparator Output	Open Collector
$\overline{G}$	Carry Generate (Active Low) Output	10 UL
$\overline{P}$	Carry Propagate (Active Low) Output	10 UL
C_4	Carry Output	10 UL

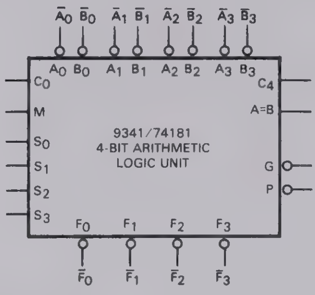
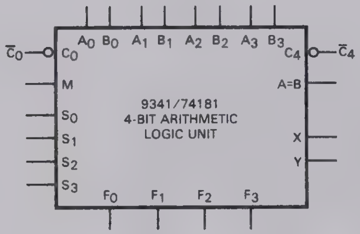
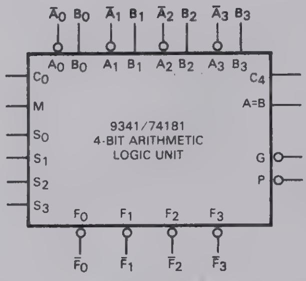
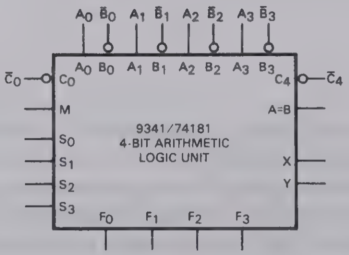
This ALU is a parallel 4-bit MSI device that can perform 16 arithmetic and all 16 possible logic operations on two 4-bit parallel words. The significant arithmetic operations are add, subtract, pass, increment, decrement, invert, and double. The operation is selected by four select lines $S_0 - S_3$ and a mode control line M, which is Low for arithmetic operations and High for logic operations. The device has a Carry In, a Carry Out for ripple carry cascading of units, and two look-ahead auxiliary carry functions, Carry Generate and Carry Propagate for use with the carry lookahead 9342. An open collector A = B output is also provided that can be AND-tied to the A = B outputs of other ALUs to detect an all High output condition for several units.

9341 OPERATION



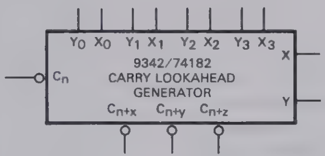
In the logic of this ALU, four identical AND/OR networks gate the A and B input operands with the four select lines S_0 – S_3 to produce the required first level auxiliary AND and OR functions. These are then used to generate the sum and carry functions. Internal carry lookahead gives high speed. The A = B output is generated by sensing the all one condition at the F outputs. When control M is in the High state, carries are inhibited from propagating and logic functions are generated at the outputs. The functions available with the device form a closed set such that inversion of the logic inputs produces a function which is still in the set. Therefore, the device performs the same logic and arithmetic functions in the active High representation as it does in the active Low representation, but with a different select code. If a mixed representation is employed, the majority of useful functions are still available. The four modes of ALU use are shown above. The operation tables for each mode are shown on the next page.

9341 OPERATION TABLES

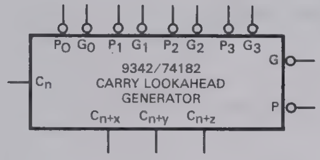
	S_0	S_1	S_2	S_3	Logic (M = H)	Arithmetic (M = L, C_O = Inactive)	Arithmetic (M = L, C_O = Active)
 $\overline{A}_0$ $\overline{B}_0$ $\overline{A}_1$ $\overline{B}_1$ $\overline{A}_2$ $\overline{B}_2$ $\overline{A}_3$ $\overline{B}_3$ A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 C_0 C_4 M A=B S_0 S_1 S_2 S_3 9341/74181 4-BIT ARITHMETIC LOGIC UNIT G P F_0 F_1 F_2 F_3 $\overline{F}_0$ $\overline{F}_1$ $\overline{F}_2$ $\overline{F}_3$	L	L	L	L	$\overline{A}$	A minus 1	A
	H	L	L	L	$\overline{A \bullet B}$	A•B minus 1	A•B
	L	H	L	L	$\overline{A} + B$	$A \bullet \overline{B}$ minus 1	$A \bullet \overline{B}$
	H	H	L	L	Logic '1'	minus 1 (2s comp.)	Zero
	L	L	H	L	$\overline{A + B}$	A plus $(A + \overline{B})$	A plus $(A + \overline{B})$ plus 1
	H	L	H	L	$\overline{B}$	A•B plus $(A + \overline{B})$	A•B plus $(A + \overline{B})$ plus 1
	L	H	H	L	$\overline{A \oplus B}$	A minus B minus 1	A minus B
	H	H	H	L	$A + \overline{B}$	$A + \overline{B}$	$A + \overline{B}$ plus 1
	L	L	L	H	$\overline{A} \bullet B$	A plus (A + B)	A plus (A + B) plus 1
	H	L	L	H	A ⊕ B	A plus B	A plus B plus 1
	L	H	L	H	B	$A \bullet \overline{B}$ plus (A + B)	$A \bullet \overline{B}$ plus (A + B) plus 1
	H	H	L	H	A + B	A + B	A + B plus 1
	L	L	H	H	Logic '0'	A plus A (2 x A)	A plus A (2 x A) plus 1
	H	L	H	H	$A \bullet \overline{B}$	A plus A•B	A plus A•B plus 1
	L	H	H	H	A•B	A plus $A \bullet \overline{B}$	A plus $A \bullet \overline{B}$ plus 1
	H	H	H	H	A	A	A plus 1
 $\overline{C}_0$ A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 C_0 C_4 $\overline{C}_4$ M A=B S_0 S_1 S_2 S_3 9341/74181 4-BIT ARITHMETIC LOGIC UNIT X Y F_0 F_1 F_2 F_3	L	L	L	L	$\overline{A}$	A	A plus 1
	H	L	L	L	$\overline{A + B}$	A + B	A + B plus 1
	L	H	L	L	$\overline{A} \bullet B$	$A + \overline{B}$	$A + \overline{B}$ plus 1
	H	H	L	L	Logic '0'	minus 1 (2s comp.)	Zero
	L	L	H	L	$\overline{A \bullet B}$	A plus $A \bullet \overline{B}$	A plus $A \bullet \overline{B}$ plus 1
	H	L	H	L	$\overline{B}$	$A \bullet \overline{B}$ plus(A + B)	$A \bullet \overline{B}$ plus (A + B) plus 1
	L	H	H	L	A ⊕ B	A minus B minus 1	A minus B
	H	H	H	L	$A \bullet \overline{B}$	$A \bullet \overline{B}$ minus 1	$A \bullet \overline{B}$
	L	L	L	H	$\overline{A} + B$	A plus A•B	A plus A•B plus 1
	H	L	L	H	$\overline{A \oplus B}$	A plus B	A plus B plus 1
	L	H	L	H	B	A•B plus $(A + \overline{B})$	A•B plus $(A + \overline{B})$ plus 1
	H	H	L	H	A•B	A•B minus 1	A•B
	L	L	H	H	Logic '1'	A plus A (2 x A)	A plus A (2 x A) plus 1
	H	L	H	H	$A + \overline{B}$	A plus (A + B)	A plus (A + B) plus 1
	L	H	H	H	A + B	A plus $(A + \overline{B})$	A plus $(A + \overline{B})$ plus 1
	H	H	H	H	A	A minus 1	A
 $\overline{A}_0$ B_0 $\overline{A}_1$ B_1 $\overline{A}_2$ B_2 $\overline{A}_3$ B_3 A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 C_0 C_4 M A=B S_0 S_1 S_2 S_3 9341/74181 4-BIT ARITHMETIC LOGIC UNIT G P F_0 F_1 F_2 F_3 $\overline{F}_0$ $\overline{F}_1$ $\overline{F}_2$ $\overline{F}_3$	L	L	L	L	$\overline{A}$	A minus 1	A
	H	L	L	L	$\overline{A} + B$	$A \bullet \overline{B}$ minus 1	$A \bullet \overline{B}$
	L	H	L	L	$\overline{A \bullet B}$	A•B minus 1	A•B
	H	H	L	L	Logic '1'	minus 1 (2s comp.)	Zero
	L	L	H	L	$\overline{A} \bullet B$	A plus (A + B)	A plus (A + B) plus 1
	H	L	H	L	B	$A \bullet \overline{B}$ plus (A + B)	$A \bullet \overline{B}$ plus (A + B) plus 1
	L	H	H	L	A ⊕ B	A plus B	A plus B plus 1
	H	H	H	L	A + B	A + B	A + B plus 1
	L	L	L	H	$\overline{A + B}$	A plus $(A + \overline{B})$	A plus $(A + \overline{B})$ plus 1
	H	L	L	H	$\overline{A \oplus B}$	A minus B minus 1	A minus B
	L	H	L	H	$\overline{B}$	A•B plus $(A + \overline{B})$	A•B plus $(A + \overline{B})$ plus 1
	H	H	L	H	$A + \overline{B}$	$A + \overline{B}$	$A + \overline{B}$ plus 1
	L	L	H	H	Logic '0'	A plus A (2 x A)	A plus A (2 x A) plus 1
	H	L	H	H	A•B	A plus $A \bullet \overline{B}$	A plus $A \bullet \overline{B}$ plus 1
	L	H	H	H	$A \bullet \overline{B}$	A plus A•B	A plus A•B plus 1
	H	H	H	H	A	A	A plus 1
 A_0 $\overline{B}_0$ A_1 $\overline{B}_1$ A_2 $\overline{B}_2$ A_3 $\overline{B}_3$ $\overline{C}_0$ A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 C_0 C_4 $\overline{C}_4$ M A=B S_0 S_1 S_2 S_3 9341/74181 4-BIT ARITHMETIC LOGIC UNIT X Y F_0 F_1 F_2 F_3	L	L	L	L	$\overline{A}$	A	A plus 1
	H	L	L	L	$\overline{A} \bullet B$	$A + \overline{B}$	$A + \overline{B}$ plus 1
	L	H	L	L	$\overline{A + B}$	A + B	A + B plus 1
	H	H	L	L	Logic '0'	minus 1 (2s comp.)	Zero
	L	L	H	L	$\overline{A} + B$	A plus A•B	A plus A•B plus 1
	H	L	H	L	B	A•B plus $(A + \overline{B})$	A•B plus $(A + \overline{B})$ plus 1
	L	H	H	L	$\overline{A \oplus B}$	A plus B	A plus B plus 1
	H	H	H	L	A•B	A•B minus 1	A•B
	L	L	L	H	$\overline{A \bullet B}$	A plus $A \bullet \overline{B}$	A plus $A \bullet \overline{B}$ plus 1
	H	L	L	H	A ⊕ B	A minus B minus 1	A minus B
	L	H	L	H	$\overline{B}$	$A \bullet \overline{B}$ plus (A + B)	$A \bullet \overline{B}$ plus (A + B) plus 1
	H	H	L	H	$A \bullet \overline{B}$	$A \bullet \overline{B}$ minus 1	$A \bullet \overline{B}$
	L	L	H	H	Logic '1'	A plus A (2 x A)	A plus A (2 x A) plus 1
	H	L	H	H	A + B	A plus $(A + \overline{B})$	A plus $(A + \overline{B})$ plus 1
	L	H	H	H	$A + \overline{B}$	A plus (A + B)	A plus (A + B) plus 1
	H	H	H	H	A	A minus 1	A

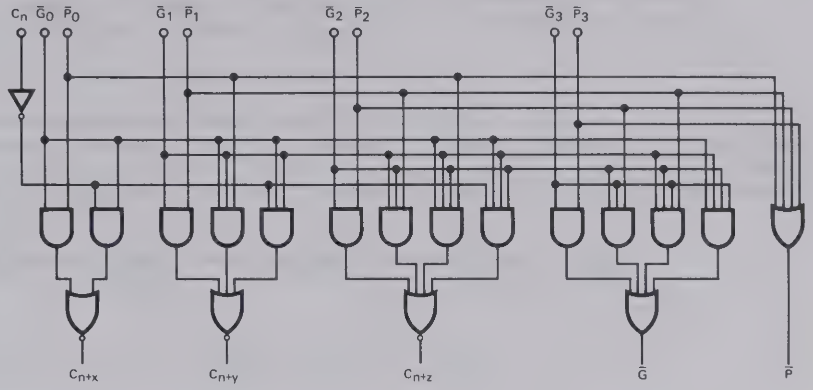
9342 CARRY LOOKAHEAD GENERATOR

ACTIVE HIGH


Y0 X0 Y1 X1 Y2 X2 Y3 X3
9342/74182
CARRY LOOKAHEAD
GENERATOR
Cn
X
Y
Cn+x Cn+y Cn+z

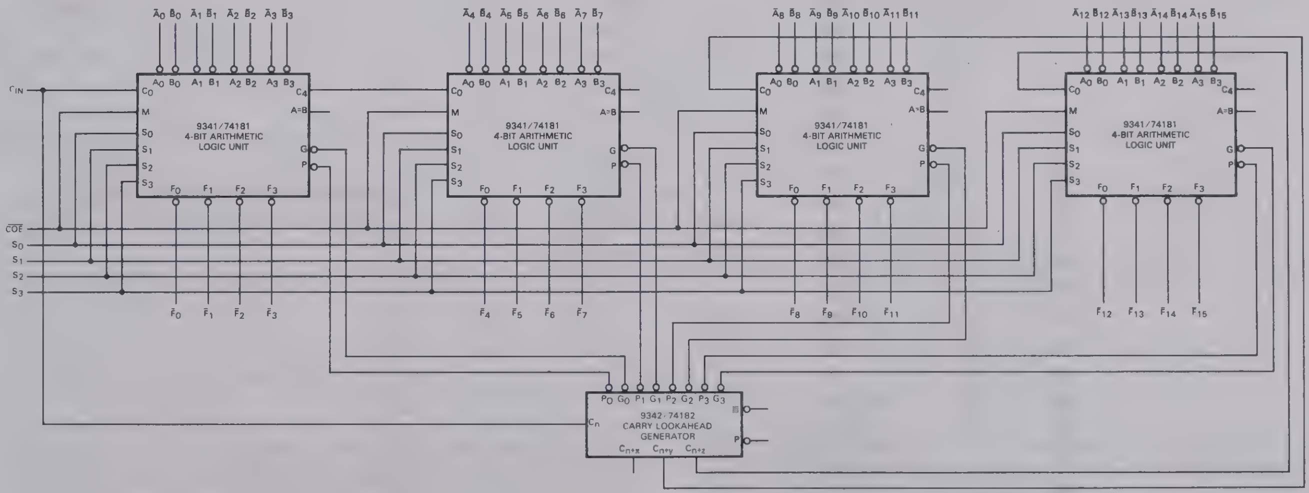
ACTIVE LOW


P0 G0 P1 G1 P2 G2 P3 G3
9342/74182
CARRY LOOKAHEAD
GENERATOR
Cn
G
P
Cn+x Cn+y Cn+z


Cn G̅0 P̅0
G̅1 P̅1
G̅2 P̅2
G̅3 P̅3
Cn+x
Cn+y
Cn+z
G̅
P̅

The 9341 ALU can be used in a variety of carry modes. The simplest of these is in a ripple carry mode where the Carry In C_{in} of an ALU is driven by the Carry Out signal C_4 from the previous ALU. This method of propagating the carry is slow for large word lengths but has the advantage that additional carry circuits are not required; if several levels of lookahead are permitted and extra logic is used, the speed of the ALU can be improved. The 9341 gives the auxiliary carry functions Carry Generate and Carry Propagate which can be used with the 9342 to give complete carry lookahead or ripple block lookahead. In this latter mode, the ALU is split into 16-bit blocks, each with its own lookahead with carries allowed to ripple between the blocks. The 9342 accepts up to four sets of Carry Generate and Carry Propagate functions and a Carry In and provides the three Carry Out signals required by the ALUs and also the next level auxiliary functions. These auxiliary functions generated by the carry lookahead circuit allow further levels of lookahead. Unfortunately, to satisfy signal polarities, a penalty of two gate delays is incurred for each level of lookahead, and the auxiliary functions are rarely used over more than two levels of lookahead. The logic symbols and logic diagram of the 9342 carry lookahead circuit are shown above. The auxiliary logic functions in the active High case are not Carry Generate and Carry Propagate, they have been labeled X and Y, respectively. Of course, they are connected in the same manner as the active Low case. In this logic design, the auxiliary functions are used to generate the three Carry Out signals and the two auxiliary functions required for further levels of lookahead.

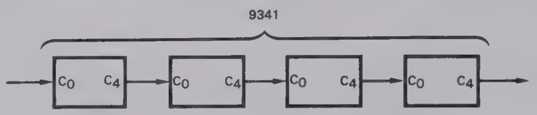
CARRY LOOKAHEAD


A̅0 B̅0 A̅1 B̅1 A̅2 B̅2 A̅3 B̅3
A̅4 B̅4 A̅5 B̅5 A̅6 B̅6 A̅7 B̅7
A̅8 B̅8 A̅9 B̅9 A̅10 B̅10 A̅11 B̅11
A̅12 B̅12 A̅13 B̅13 A̅14 B̅14 A̅15 B̅15
CIN
C̅O̅E̅
S0
S1
S2
S3
9341/74181
4-BIT ARITHMETIC
LOGIC UNIT
C0
M
C4
A=B
G
P
F0 F1 F2 F3
F̅0 F̅1 F̅2 F̅3
F̅4 F̅5 F̅6 F̅7
F̅8 F̅9 F̅10 F̅11
F̅12 F̅13 F̅14 F̅15
P0 G0 P1 G1 P2 G2 P3 G3
9342/74182
CARRY LOOKAHEAD
GENERATOR
Cn
Cn+x Cn+y Cn+z

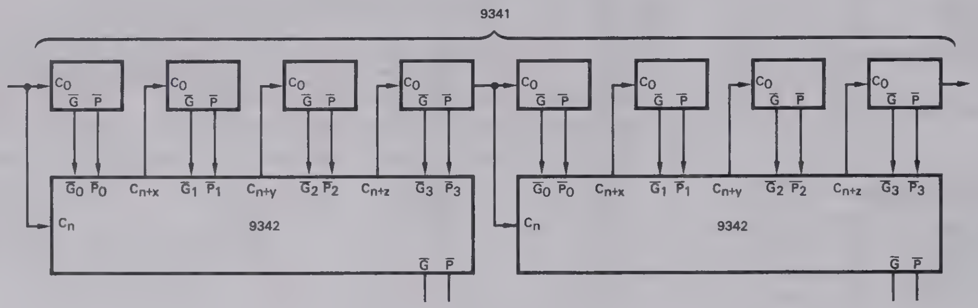
A single carry lookahead circuit is used with four 9341 ALUs to perform arithmetic operations with complete carry lookahead over 16-bit words. For word lengths of 20 and 24 bits, the fastest speed is achieved by only using a single 9342 as above and letting the carry ripple through the additional one or two 9341s. For word lengths of 28 and 32 bits, the fastest speed is achieved by using two 9342s, constructing two blocks similar to the 16-bit block above, and letting the carry ripple from the first block to the second. Only when the word length exceeds 32 bits is there a speed advantage in using three levels of carry lookahead.

9341 CARRY METHODS

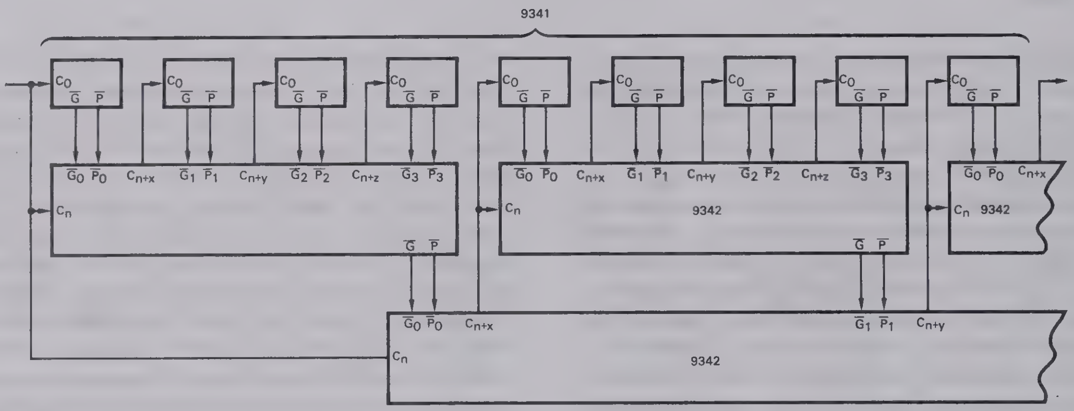
SINGLE LEVEL (INTERNAL) CARRY LOOKAHEAD:


9341
C0 C4
C0 C4
C0 C4
C0 C4

TWO LEVEL CARRY LOOKAHEAD:


9341
C0 G P
G0 P0 Cn+x G1 P1 Cn+y G2 P2 Cn+z G3 P3
Cn
9342
G P

THREE LEVEL CARRY LOOKAHEAD:


9341
C0 G P
G0 P0 Cn+x G1 P1 Cn+y G2 P2 Cn+z G3 P3
Cn
9342
G P
G0 P0 Cn+x G1 P1 Cn+y

No. of Bits	Total Addition Time (ns)	Add Time Per Bit (ns)	Package Count 9341	Package Count 9342
4	21	5.2	1	
8	36	4.5	2	
12	48	4.0	3	
12	36	3.0	3	1
16	60	3.7	4	
16	36	2.2	4	1
32	110	3.4	8	
32	86	2.7	8	1
32	62	1.9	8	2
48	160	3.3	12	
48	136	2.8	12	1
48	112	2.3	12	2
48	88	1.8	12	3
48	64	1.3	12	4
64	210	3.3	16	
64	162	2.5	16	2
64	138	2.2	16	3
64	101	1.6	16	4
64	64	1.0	16	5

There are various methods of using the 9342 with ALUs to perform arithmetic operations over large word lengths. The table above gives typical delays realized from various connection methods. It shows how the use of a small number of carry lookahead packages considerably decreases the delay for typical word lengths used in digital systems. The addition delay times for various configurations are given. Subtraction times are approximately 3 ns longer.

ARITHMETIC WITH THE 9341

The 9341 can be used as an arithmetic element in all the common binary number representations. The basic concepts as discussed for the 9340 apply also to the 9341. The most difficult number representation is sign magnitude. The 9341 is more flexible than the 9340 but additional peripheral logic must be used to decode the desired functions required by the select lines, and the Carry In at the first stage. The two most useful additional arithmetic operations that the 9341 has as compared to the 9340 are "double A" multiplication, division and square root operations. It is often possible to use the select code on the ALU to effectively perform additional decoding, for example if a control signal is to select between "add A and B" when S is High and "pass A" when S is Low, then (for the active High case) S_0 is tied to S_3 to form S, and S_1 and S_2 are tied Low. This type of operation (Add or Pass) is useful in multiplication routines.

COMPARISON WITH THE 9341

Output	State	Operation	Active LOW Logic	Active HIGH Logic
A=B	H	A minus B	A=B	A = (B minus 1)
	H	$\overline{A \oplus B}$	$A \not\equiv B$	A=B
	H	$A \oplus B$	A=B	$A \not\equiv B$
Carry Out	H	A minus B	$A \geqslant B$	$A < B$
($\overline{C}_4$ for active High operands)	L	A minus B	$A < B$	$A \geqslant B$
(C_4 for active Low operands)	H	A minus B minus 1	$A > B$	$A \leqslant B$
	L	A minus B minus 1	$A \leqslant B$	$A > B$

Several comparison functions can be performed with the 9341 by using the A = B and the C_4 outputs. The A = B output is better described as "F = 0," since this output goes High anytime all the F outputs are High. Therefore, the outputs are not only used for comparing A = B during a subtract operation, but also to ascertain that the function outputs are all High after any arithmetic or logic operation. In the PASS operation, the output indicates that one of the operands is equal to zero. In the Exclusive-OR operation, it indicates that the two operands are identical. In the EQUIVALENCE operation, it indicates that the two operands are complementary. For unsigned numbers with the most significant bit positive, the Carry Out of the ALU indicates relative magnitude. This table lists the various comparison functions which can be performed in active High and active Low logic.

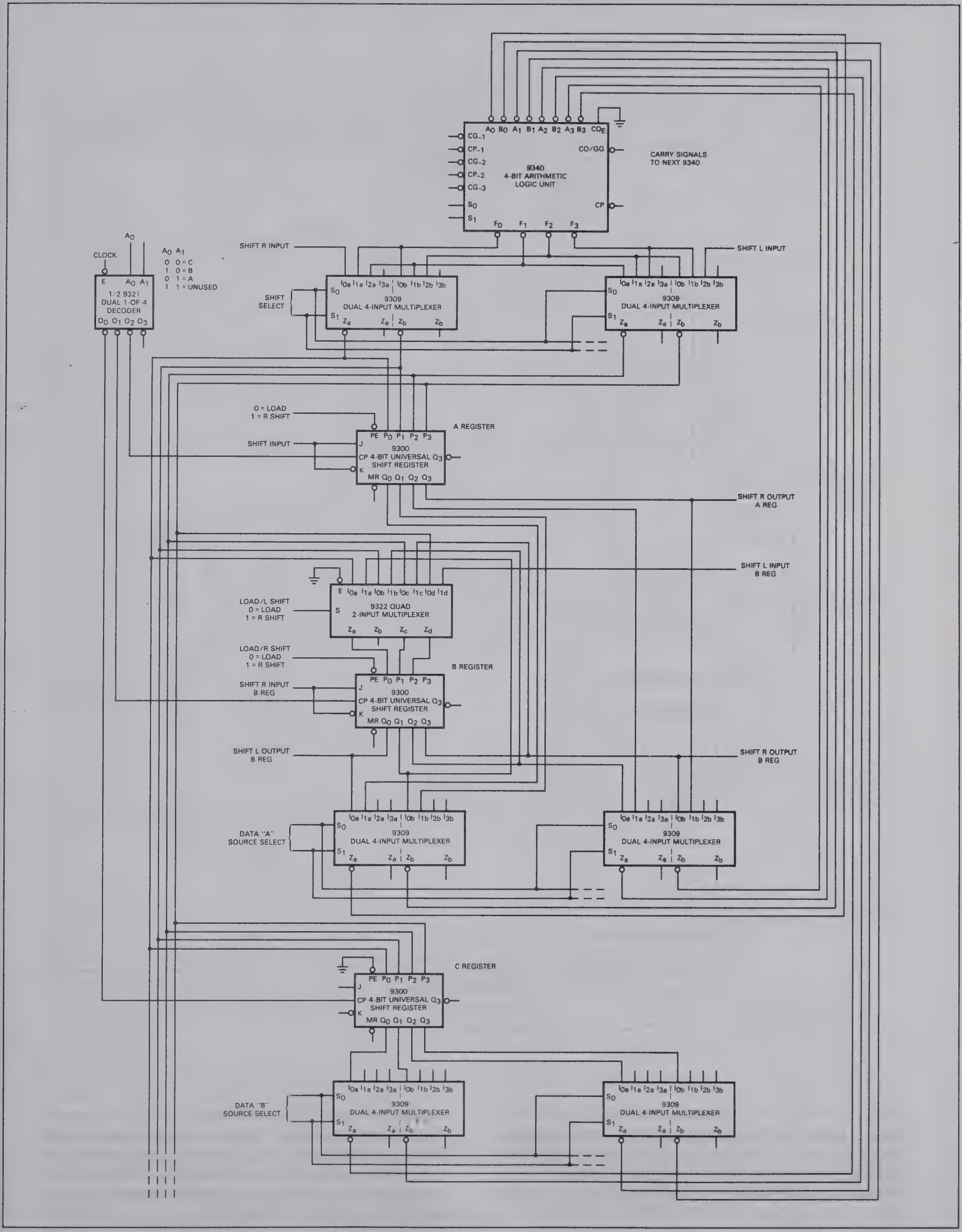
ARITHMETIC REGISTER SYSTEMS

In most digital machines, the Arithmetic Logic Unit is part of a data processing section which includes one or more data storage registers and some multiplexers for data routing, shifting, etc. Because different MSI functions work together so well in this application, it is important to consider several such systems. The most useful components for these systems, besides the 9340 and 9341 ALUs are the 9300 universal register, the 9338 multiple-port register, and the 9309 dual 4-input multiplexer. The 9338 is used in many mini computer applications because it provides eight accumulators in a convenient 3-address structure in one package. The systems shown can use either the 9340 or the 9341, though for illustration purposes only one of the two devices is shown.

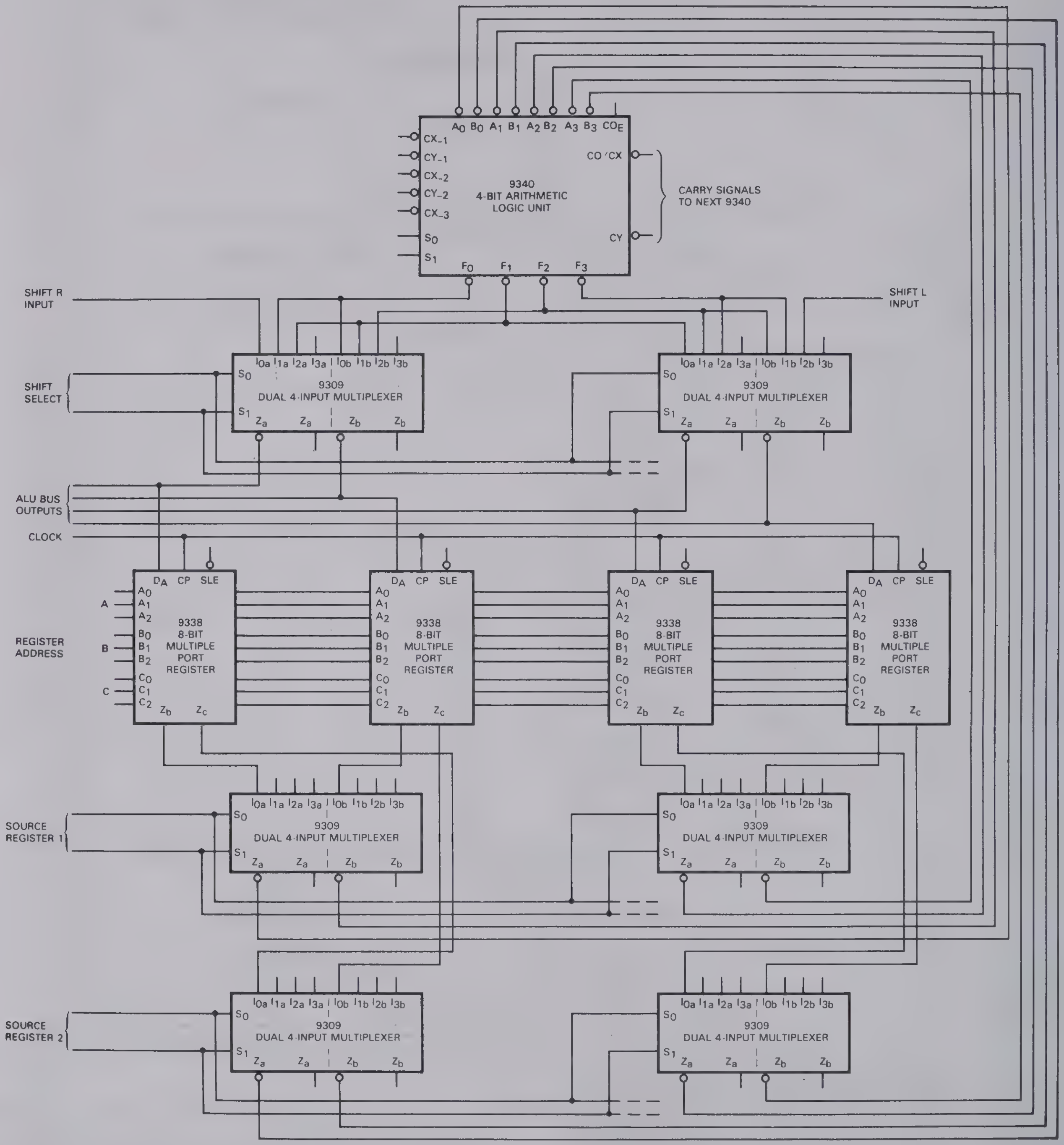
SINGLE ADDRESS ARITHMETIC REGISTER

The 4-bit slice of a single address arithmetic register shown opposite incorporates a 9340 arithmetic logic unit, 9309 multiplexers for shifting capability, and 9300s acting as working registers A, B, and C. Double length is possible by shifting the operand through the adder and the other half through the 9322 associated with the second working register. A third register, C, allows multiplication and division. This register normally would hold the divisor in division and the multiplicand in multiplication. The 9340 ALU is very useful in conjunction with the 9338 multiple-port register for processing systems with 2, and with 2 or 3-address formats.

SINGLE ADDRESS ARITHMETIC REGISTER


A0 B0 A1 B1 A2 B2 A3 B3 COE
CG-1
CP-1
CG-2
CP-2
CG-3
S0
S1
CO/GG
CP
9340
4-BIT ARITHMETIC
LOGIC UNIT
F0 F1 F2 F3
CARRY SIGNALS
TO NEXT 9340
A0
CLOCK
A0 A1
0 0 = C
1 0 = B
0 1 = A
1 1 = UNUSED
E A0 A1
1/2 9321
DUAL 1-OF-4
DECODER
O0 O1 O2 O3
SHIFT R INPUT
SHIFT L INPUT
SHIFT SELECT
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309
DUAL 4-INPUT MULTIPLEXER
Za Za Zb Zb
0 = LOAD
1 = R SHIFT
A REGISTER
SHIFT INPUT
PE P0 P1 P2 P3
J
CP
K
9300
4-BIT UNIVERSAL
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
SHIFT R OUTPUT
A REG
SHIFT L INPUT
B REG
E I0a I1a I0b I1b I0c I1c I0d I1d
S
9322 QUAD
2-INPUT MULTIPLEXER
Za Zb Zc Zd
LOAD/L SHIFT
0 = LOAD
1 = R SHIFT
LOAD/R SHIFT
0 = LOAD
1 = R SHIFT
B REGISTER
SHIFT R INPUT
B REG
SHIFT L OUTPUT
B REG
SHIFT R OUTPUT
B REG
DATA "A"
SOURCE SELECT
C REGISTER
DATA "B"
SOURCE SELECT

3-ADDRESS ARITHMETIC REGISTER

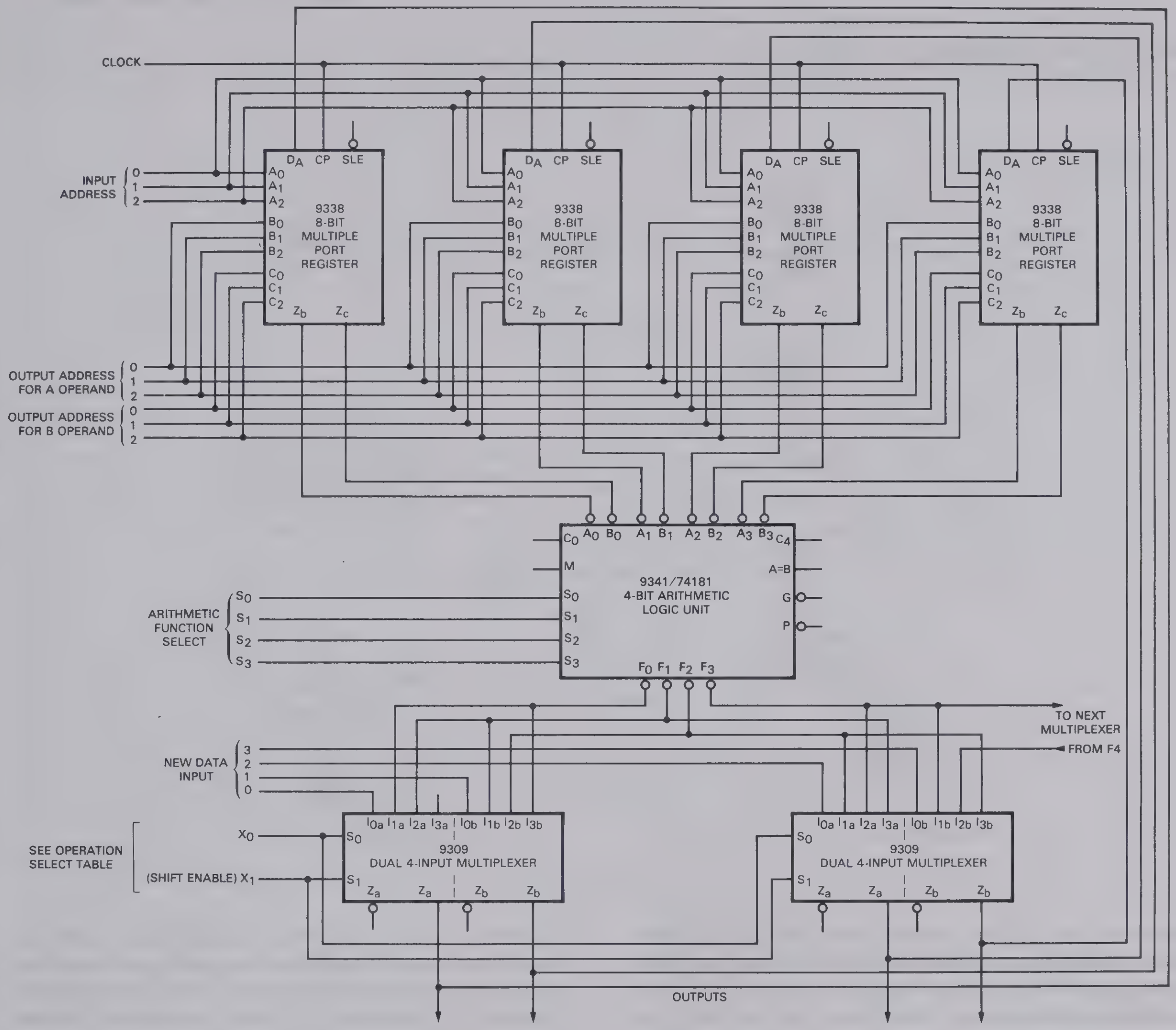


A0 B0 A1 B1 A2 B2 A3 B3 COE
CX-1
CY-1
CX-2
CY-2
CX-3
S0
S1
F0 F1 F2 F3
CO/CX
CY
9340
4-BIT ARITHMETIC
LOGIC UNIT
CARRY SIGNALS
TO NEXT 9340
SHIFT R
INPUT
SHIFT L
INPUT
SHIFT
SELECT
9309
DUAL 4-INPUT MULTIPLEXER
I0a I1a I2a I3a I0b I1b I2b I3b
Za Za Zb Zb
ALU BUS
OUTPUTS
CLOCK
DA CP SLE
9338
8-BIT
MULTIPLE
PORT
REGISTER
A0 A1 A2 B0 B1 B2 C0 C1 C2
Zb Zc
A
B
C
REGISTER
ADDRESS
SOURCE
REGISTER 1
SOURCE
REGISTER 2

This is a 4-bit slice of a typical 3-address arithmetic register system where two operands can be taken from any two registers, operated upon, and the result written into any register in the system. Three addresses are required, two for the source operands and one for the result. The unit can perform arithmetic, shifting, and logical operations in a 3-address format. By appropriate microinstructions, two consecutive register addresses can be used as a double-length register for multiplication and division. Provision is made by 9309 multiplexers in the design to allow peripheral and storage inputs to be sent to the ALU.

ADDERS/SUBTRACTORS

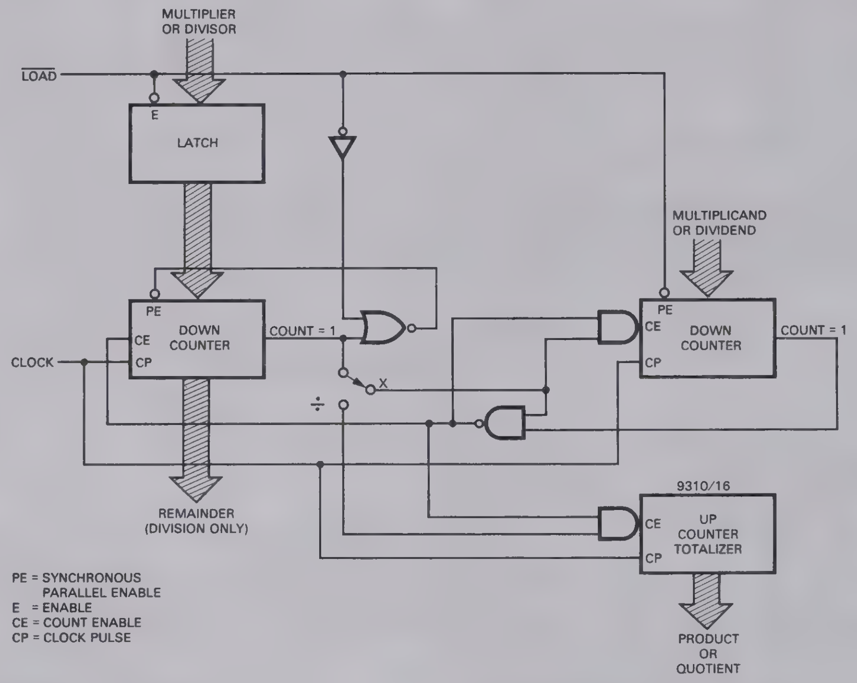
SECTION OF 3-ADDRESS ARITHMETIC LOGIC UNIT WITH EIGHT REGISTERS


CLOCK
INPUT ADDRESS 0 1 2
DA CP SLE
A0 A1 A2 B0 B1 B2 C0 C1 C2
9338 8-BIT MULTIPLE PORT REGISTER
Zb Zc
OUTPUT ADDRESS FOR A OPERAND 0 1 2
OUTPUT ADDRESS FOR B OPERAND 0 1 2
C0 A0 B0 A1 B1 A2 B2 A3 B3 C4
M
S0 S1 S2 S3
9341/74181 4-BIT ARITHMETIC LOGIC UNIT
A=B
G
P
F0 F1 F2 F3
ARITHMETIC FUNCTION SELECT S0 S1 S2 S3
TO NEXT MULTIPLEXER
FROM F4
NEW DATA INPUT 3 2 1 0
SEE OPERATION SELECT TABLE
X0
(SHIFT ENABLE) X1
I0a I1a I2a I3a I0b I1b I2b I3b
S0 S1
9309 DUAL 4-INPUT MULTIPLEXER
Za Za Zb Zb
OUTPUTS

X_0	X_1	OPERATION
L	L	LOAD NEW DATA
H	L	NO SHIFT
L	H	SHIFT LEFT
H	H	SHIFT RIGHT

A slightly simpler system using the 9341 ALU is shown. The 9338 provides eight working registers in a 3-address format, as before. Only one multiplexer is used in the loop, instead of three. This means the outputs of the 9341 can be shifted left, right, or remain unchanged before being returned to the 9338. There is only one location for new data to enter the system. The L-L code on the multiplexer's select inputs routes data from the external source to the inputs of the 9338.

MULTIPLICATION/DIVISION USING COUNTERS



PE = SYNCHRONOUS PARALLEL ENABLE
E = ENABLE
CE = COUNT ENABLE
CP = CLOCK PULSE

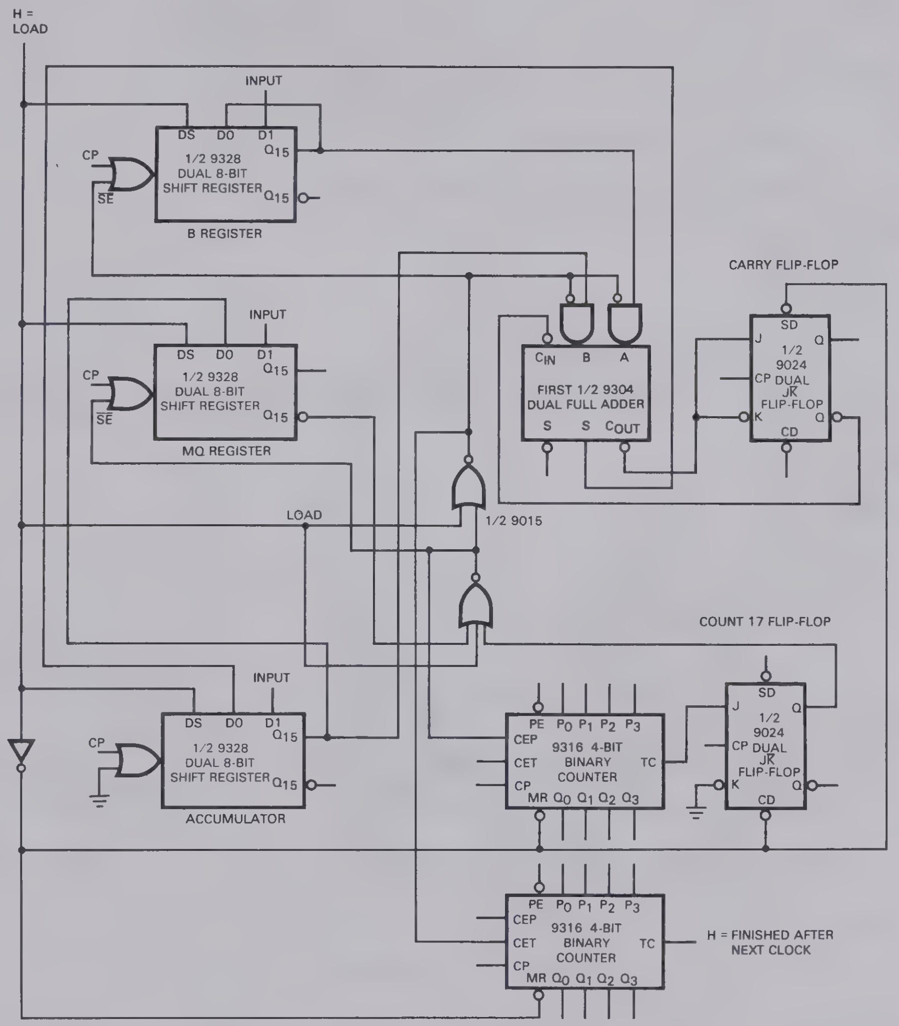
Many digital systems perform multiplication and division; computers generally break these operations down into simpler functions such as add, subtract or shift and existing arithmetic logic units (ALUs) can perform these operations under program control. In these systems, multiplication and division are actually performed by software.

However, there are systems requiring special hardware to perform multiplication and division. These are either small, simple systems in which overhead (program storage and control necessary for software implementation) is prohibitive, or they are sophisticated high speed systems (like Fast Fourier Transforms and other scientific real time computers) in which software multiplication and division is too time consuming.

Multiplication and division in slow systems can be performed in several different ways. The slowest method uses counters and is particularly attractive for BCD operation or in special cases where the operands are expressed in different numbering systems. The algorithm used in a counting multiplier/divider either counts M pulses n times for multiplication or counts the number of times n can be subtracted from M for division. In the method illustrated, a switch is used to select multiplication or division. The same system can perform addition and subtraction with slight modification. During multiplication the multiplier is stored in a latch or register. The multiplicand is loaded into one of the down counters while the multiplier is loaded into the other one. Clock pulses are then fed to the multiplier (down counter) and the totalizer (up counter). The multiplier is counted down until it reaches state "1". This state is decoded and used to reload the multiplier into the counter and also to decrement the multiplicand counter by one. This process continues until both counters reach state "1". This condition is decoded, stops all counters, and generates an Output Ready signal.

Division is performed by loading the divisor into the latch and one of the down counters and loading the dividend into the other down counter. In division, the two down counters are decremented simultaneously until both reach state "1"; but every time the divisor counter reaches "1", it is reloaded and the totalizer is incremented once. Thus the totalizer counts the number of times that the divisor must be reloaded in order to exhaust the contents of the dividend counter.

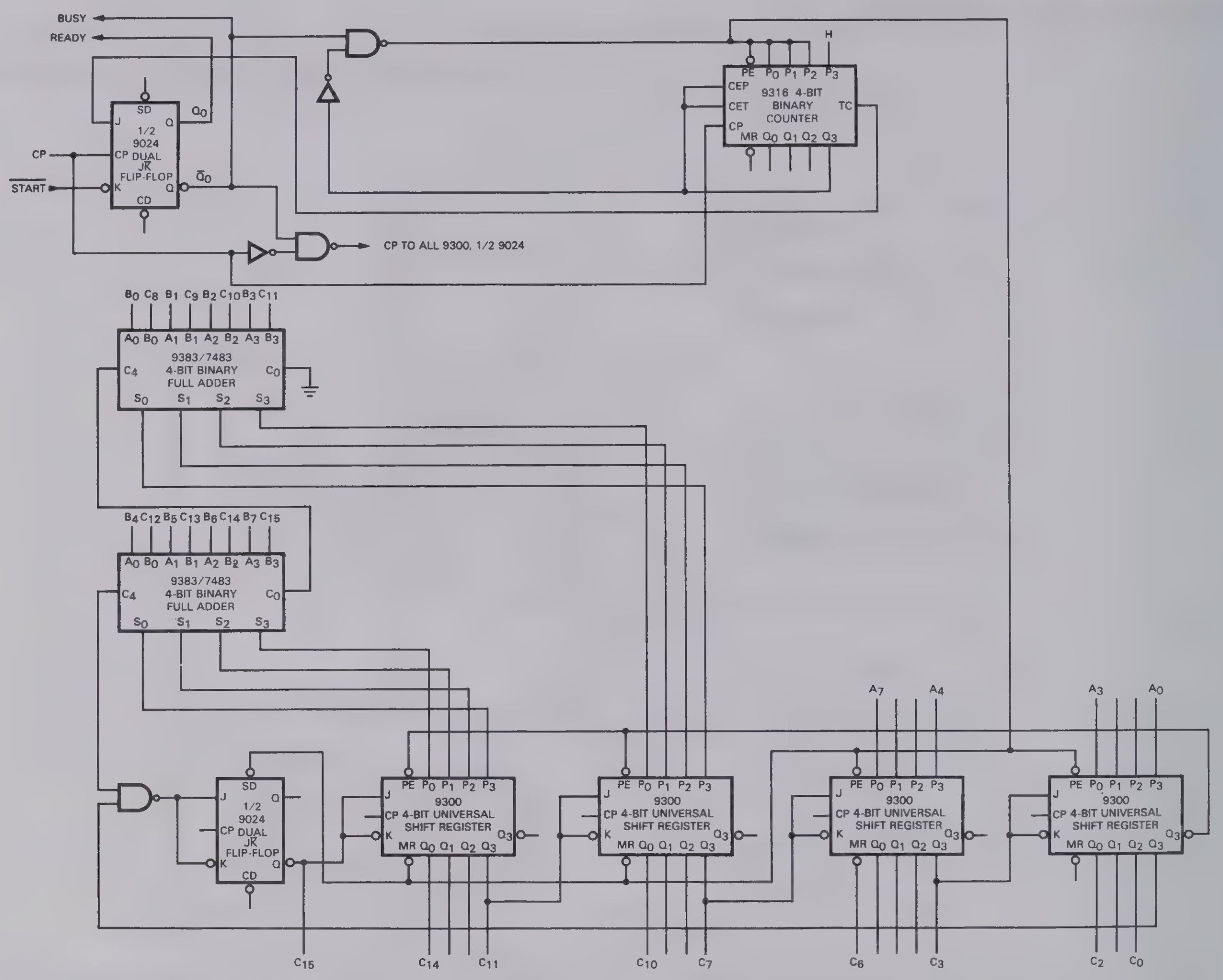
FULLY SERIAL BINARY MULTIPLIER


H = LOAD
INPUT
DS D0 D1
Q15
1/2 9328 DUAL 8-BIT SHIFT REGISTER
CP
SE
B REGISTER
MQ REGISTER
ACCUMULATOR
LOAD
CARRY FLIP-FLOP
CIN B A
FIRST 1/2 9304 DUAL FULL ADDER
S S COUT
SD J Q CP K CD
1/2 9024 DUAL JK FLIP-FLOP
1/2 9015
COUNT 17 FLIP-FLOP
PE P0 P1 P2 P3
CEP CET CP TC
9316 4-BIT BINARY COUNTER
MR Q0 Q1 Q2 Q3
H = FINISHED AFTER NEXT CLOCK

The simplest binary multiplier uses the conventional shift-and-add algorithm with a bit-serial (shift register) accumulator. Depending on the LSB of the multiplier, the multiplicand is serially added to the accumulator or is ignored; then the multiplier is shifted one bit and the process repeated. An n x n-bit multiplication requires n serial additions and approximately n^2 clock pulses; but it is possible to save almost half of these in an average multiplication.

The circuit shown is a fully serial 16 x 16-bit binary multiplier. Initially the multiplicand is stored in the B register and the multiplier in the MQ register. During the next 16 clock pulses the multiplicand is ANDed with the multiplier. This partial product is added to any contents of the accumulator and the sum is returned to the accumulator. Carries are stored in the carry flip-flop. The 17th clock pulse sets the Count 17 flip-flop. This prevents the multiplicand from shifting, but allows the accumulator and MQ register to shift once. As a result, the LSB in the accumulator enters the MQ register, the carry flip-flop contents shifts to the MSB of the accumulator, clearing the carry flip-flop. This circuit saves time by skipping zeros in the multiplier. When the LSB of the multiplier is a zero, an artificial Terminal Count is forced, saving 16 unnecessary clock pulses. An average multiplication needs 144 clock pulses; the longest multiplication requires 272 clock pulses (55 μs at 5 MHz). At the end of the multiplication, the TC of the second 9316 counter goes High. After the next clock pulse, the most significant 16 bits of the product is properly aligned in the accumulator and the least significant 16-bits of the product is in the MQ register. Operands of any length can be used by varying the shift register length and the counter modulo mode.

8 BY 8-BIT BINARY MULTIPLIER

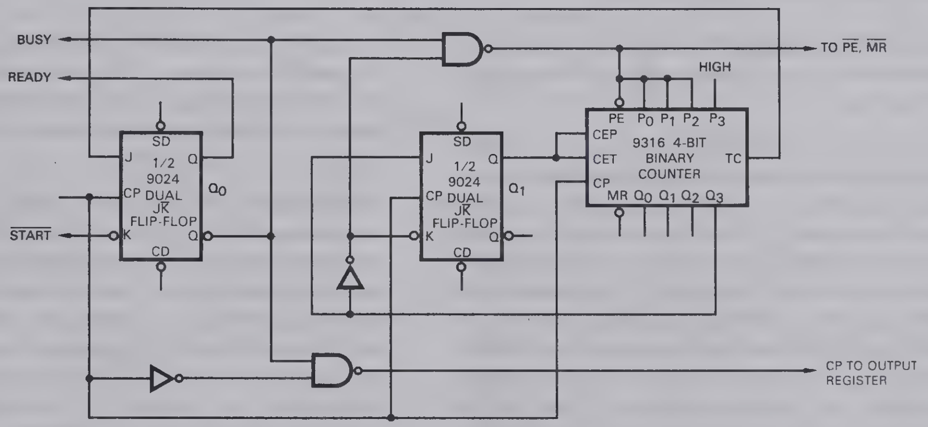

BUSY
READY
CP
START
SD
J
CP
K
CD
Q
Q0
1/2 9024 DUAL JK FLIP-FLOP
H
PE P0 P1 P2 P3
CEP
CET
CP
9316 4-BIT BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CP TO ALL 9300, 1/2 9024
B0 C8 B1 C9 B2 C10 B3 C11
A0 B0 A1 B1 A2 B2 A3 B3
9383/7483 4-BIT BINARY FULL ADDER
C4
C0
S0 S1 S2 S3
B4 C12 B5 C13 B6 C14 B7 C15
9383/7483 4-BIT BINARY FULL ADDER
9300 4-BIT UNIVERSAL SHIFT REGISTER
Q3
A7
A4
A3
A0
C15
C14
C11
C10
C7
C6
C3
C2
C0

This circuit performs the conventional shift-and-add algorithm for binary multiplication. It accepts two 8-bit words (A_{0-7} and B_{0-7}) and generates the 16-bit product C_{0-15} after 10 clock pulses. The system is self-contained, requiring a continuously running clock and generating a Ready signal that indicates when the product is available at the 16 outputs. In the Idle mode, the 9316 control counter is stopped in position 0 and the $\overline{Q_0}$ Busy output is Low, inhibiting clock pulses to the input/output register. A High-to-Low transition on the $\overline{\text{Start}}$ input initiates a multiplication. The next Low-to-High clock transition resets Q_0 (Ready), generating $\overline{Q_0}$ = Busy and enabling the clock to the input/output registers. It also activates the Parallel Enable ($\overline{\text{PE}}$) inputs of both the 9316 control counter and the A register, as well as the Master Reset ($\overline{\text{MR}}$) inputs of the remaining output register. After the next clock pulse, the 9316 control counter is loaded with code 8, the A register is loaded with the eight bits of factor A, and the remaining registers are cleared. The next eight clock pulses perform the actual multiplication. Each clock pulse does the following.

- Increments the 9316 control counter, *and*
- Right shifts the eight right-hand bits of the input/output register, *and*
- Right shifts the entire C register when the LSB of the A register is zero, *or*
- Adds factor B (B_{0-7}) to the contents of the eight leftmost positions of the C register (C_{8-15}) and inserts the sum one position further to the right when the LSB of the A register is one.

When the control counter has reached TC (position 15) it sets Q_0 which generates Ready, removes the Busy output, indicating that the product is available at the outputs C_{0-15}.

CONTROL SECTION OF A 16 BY 16-BIT BINARY MULTIPLIER


BUSY
READY
START
SD
J
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
CD
Q
Q0
Q
SD
J
1/2
9024
CP DUAL
JK
FLIP-FLOP
K
CD
Q
Q1
Q
PE P0 P1 P2 P3
CEP
CET
9316 4-BIT
BINARY
COUNTER
CP
TC
MR Q0 Q1 Q2 Q3
HIGH
TO PE, MR
CP TO OUTPUT
REGISTER

	9383		9340	
	8 x 8	16 x 16	8 x 8	16 x 16
Max. clock frequency MHz	6	3	10	10
Multiply cycle time μs	1.6	6.4	1	2
Number of components	7 MSI	13 MSI	7 MSI	13 MSI
	2¼ SSI	3 SSI	2¼ SSI	3 SSI

The binary multiplier previously described can be expanded to provide 16 x 16-bit multiplication. The control counter must be extended by adding a least significant flip-flop and the adder and register arrays must be doubled. The longer ripple delay through the 9383 adders limits the clock frequency to ≈ 3 MHz (vs ≈ 6 MHz for the 8 x 8 multiplier) resulting in a cycle time four times longer. System speed can be improved by using the faster (but larger and more expensive) 9340 carry lookahead adders.

COUNT AND SHIFT BCD MULTIPLICATION

BCD multiplication is considerably more involved than binary multiplication due to the many binary-to-BCD corrections required.

Since BCD notation is predominantly used in slow human-oriented systems, several unconventional methods are attractive which trade off longer response time for less circuit complexity. One such method already illustrated uses counters for multipliers. A faster system multiplies two 4-digit numbers to generate an 8-digit result in less than 300 μs (typically in ≈150 μs) using 16 MSI and 3 SSI devices. It performs multiplication on a digit-by-digit basis, going through 16 operations to generate the 16 partial products. Each partial product is generated by counting, but an 8-decade synchronous counter is used to accumulate and store the result using the synchronous parallel load feature to perform decimal shifts in accordance with the decimal weight of each partial product.

In the Idle mode both control flip-flops Q_0 and Q_1 are reset. $\overline{Q}_0$ loads counter D continuously forcing its Q_3 output High which inhibits shifting and counting in the result counter/register currently holding the result of the previous multiplication.

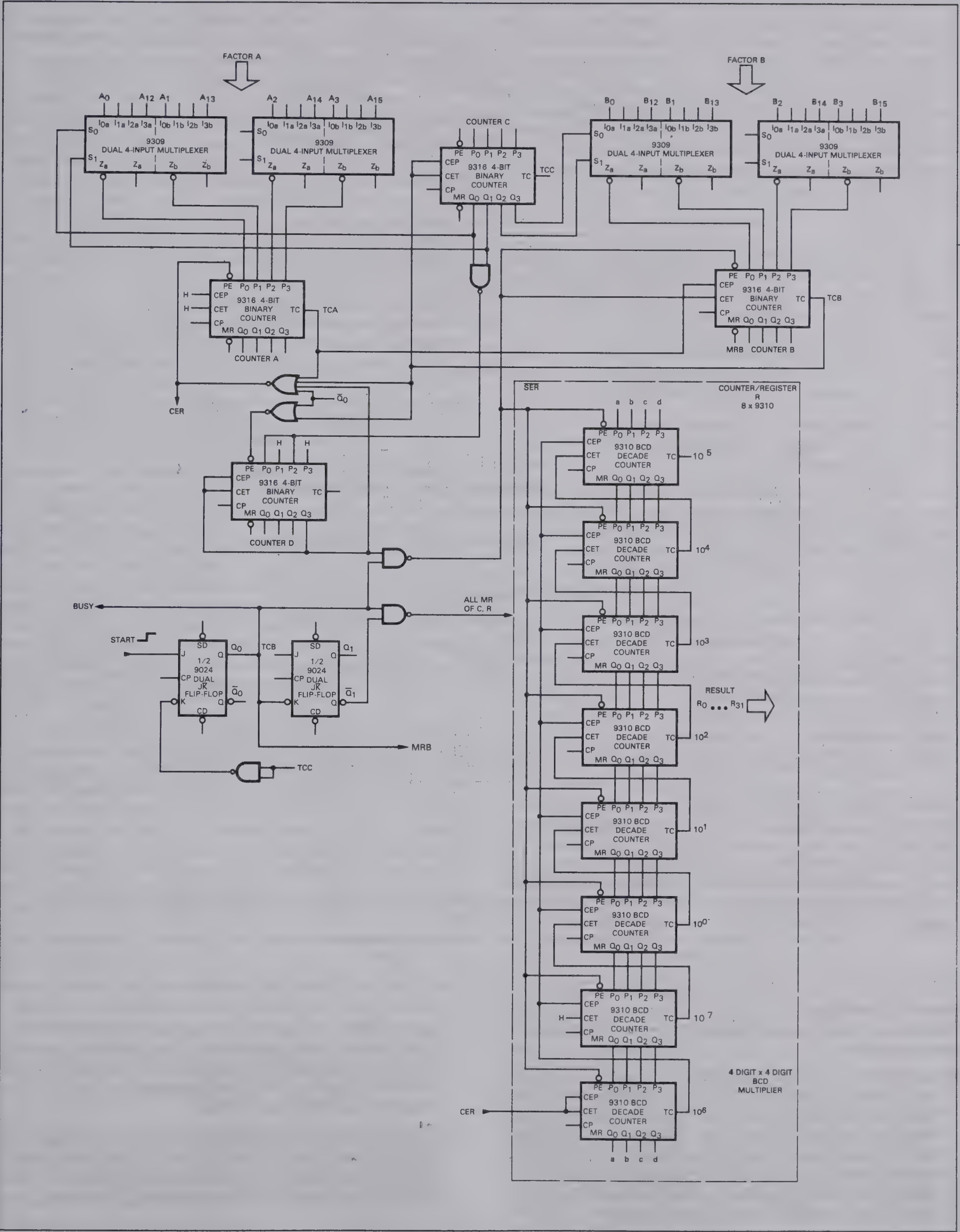
The first clock pulse (f ≈ 6 MHz) after a Low-to-High transition on the Start input sets Q_0 which activates the Busy output, allows counter D to count, clears the C and R counters, and activates the $\overline{PE}$ input of counter B. The next clock pulse advances counter D to zero (where it stays until reloaded by TCB), and lets the two counters A and B count up, starting with the complement of the least significant digits of A and B. Counters C and R, however, are still held at zero by their $\overline{MR}$ inputs. When (after < 90 clock pulses) TCB goes High, it activates $\overline{PE}$ of counter D, so that the next clock pulse loads a binary 15 into this counter. TCB also activates the J input of control flip-flop Q_1 so that the next clock pulse terminates the Clear input to counters C and R. This ends the initializing housekeeping.

When counter D was at 15, it had again loaded the 15 complement of the LSDs of factors A and B into the two counters. When counter D goes to zero, it starts the first partial multiplication; letting counter A count to TCA, reload, count to TCA, reload, etc., each TCA incrementing counter B, until TCB is reached. Counter R (the result counter) is incremented every time, except when counters A or B are at TC. When TCB is reached, it increments counter C, so that the next digit of factor A is loaded into counter A. TBC also loads counter D with a binary 15, which shifts the contents of counter R one step down thereby effectively dividing R by 10. The same clock pulse loads counters A and B and increments counter D to zero, which starts the next partial product.

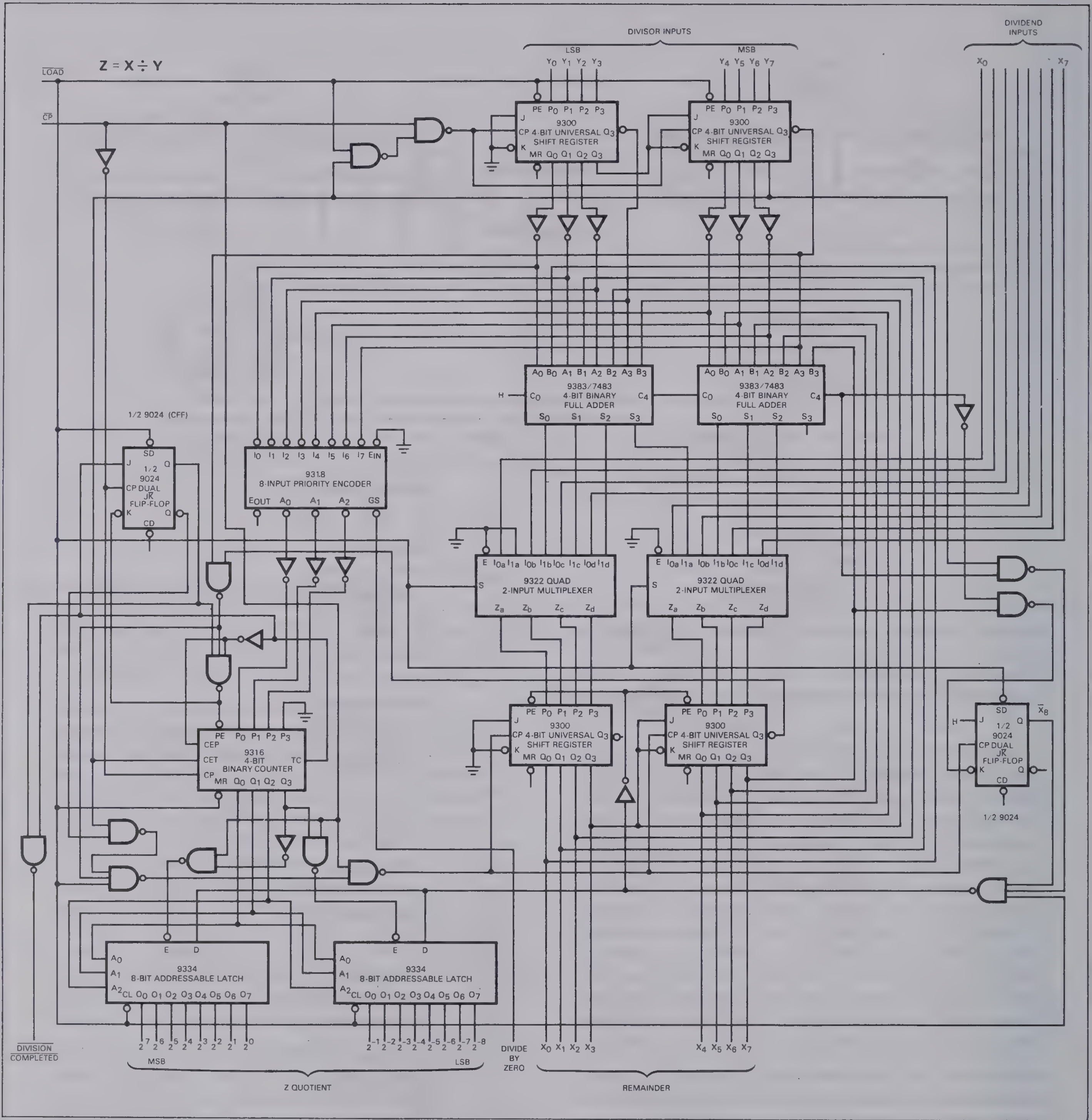
When the last digit of A has been used, it is necessary to shift the contents of counter R two positions up (instead of one position down). This is achieved by loading a binary 10 (instead of 15) into counter D, which results in six decimal down shifts, equivalent to two upshifts, since counter R shifts in a ring.

The multiplication is finished when the last TCB increments counter C from 15 to zero and generates TCC, which resets Q_0, and inhibits all operations. Counter R now contains the product in its eight decade counters. The decimal positions are indicated in the result counter/register. This circuit lends itself very readily to display multiplexing of A, B, and R, since all the necessary multiplexing hardware is already there.

COUNT AND SHIFT BCD MULTIPLICATION


FACTOR A
FACTOR B
A0 A12 A1 A13
A2 A14 A3 A15
B0 B12 B1 B13
B2 B14 B3 B15
9309 DUAL 4-INPUT MULTIPLEXER
S0 S1 I0a I1a I2a I3a I0b I1b I2b I3b Za Zb
COUNTER C
9316 4-BIT BINARY COUNTER
PE P0 P1 P2 P3 CEP CET CP TC MR Q0 Q1 Q2 Q3
TCC
TCA
TCB
COUNTER A
COUNTER B
COUNTER D
MRB
H
CER
Q̄0
SER
COUNTER/REGISTER R 8 x 9310
a b c d
9310 BCD DECADE COUNTER
10^5
10^4
10^3
10^2
10^1
10^0
10^7
10^6
RESULT
R0 ... R31
ALL MR OF C, R
BUSY
START
SD CD J K Q Q̄
1/2 9024 DUAL JK FLIP-FLOP
Q0
Q1
Q̄1
TCC
4 DIGIT x 4 DIGIT BCD MULTIPLIER

8-BIT DIVIDER


Z = X ÷ Y
LOAD
CP
DIVISOR INPUTS
LSB
MSB
Y0 Y1 Y2 Y3
Y4 Y5 Y6 Y7
DIVIDEND INPUTS
X0
X7
PE P0 P1 P2 P3
9300
CP 4-BIT UNIVERSAL Q3
SHIFT REGISTER
MR Q0 Q1 Q2 Q3
J
K
A0 B0 A1 B1 A2 B2 A3 B3
9383/7483
4-BIT BINARY
FULL ADDER
C0
C4
S0 S1 S2 S3
H
1/2 9024 (CFF)
SD
1/2
9024
CP DUAL
JK
FLIP-FLOP
CD
Q
I0 I1 I2 I3 I4 I5 I6 I7 EIN
9318
8-INPUT PRIORITY ENCODER
EOUT A0 A1 A2 GS
E I0a I1a I0b I1b I0c I1c I0d I1d
9322 QUAD
2-INPUT MULTIPLEXER
S
Za Zb Zc Zd
CEP
CET
9316
4-BIT
BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
X̄8
1/2 9024
E D
A0
A1
A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
DIVISION COMPLETED
2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0
2^-1 2^-2 2^-3 2^-4 2^-5 2^-6 2^-7 2^-8
MSB
LSB
Z QUOTIENT
DIVIDE BY ZERO
X0 X1 X2 X3
X4 X5 X6 X7
REMAINDER

BINARY DIVISION

Division is a more complex operation than multiplication for several reasons.

- Multiplication is usually performed by a sequence of shift and conditional add operations, where the bits of the multiplier determine whether or not the multiplicand should be added into the accumulator. Division, on the other hand, is usually performed by a sequence of shift and conditional *subtract* operations, but the decision about subtracting the divisor from the dividend is based on whether or not this subtraction would cause underflow. This requires either a magnitude comparator, or a trial subtraction, followed when necessary by an addition to cancel it (restoring method), or it requires a more sophisticated control sequence, alternating between subtraction and addition (non-restoring method).
- While the result of a multiplication is always as long as both factors put end to end, division in most cases generates a result that is an infinite stream of bits and must therefore be cut-off by either truncating or rounding.

- Moreover, while any number can be multiplied by any other number generating a meaningful and defined result, division by zero requires special consideration; dividing a non zero number by zero results in infinity (overflow), and dividing zero by zero results in an undefined answer.

Most digital systems perform division by a sequence of subtract, test, and shift operations controlled by software and employing the same Arithmetic Logic Unit normally used for addition and subtraction. However, the addition of special, hard-wired circuits for the sole purpose of performing division is sometimes required in systems which are either too small to have an ALU, or so large and fast that several operations must be performed simultaneously in the shortest possible time.

The opposite page shows an example of such a hardware divider, an 8-bit binary divider built with MSI devices. This circuit accepts an 8-bit dividend ($X_0 - X_7$) and an 8-bit divisor ($Y_0 - Y_7$) and generates a 16-bit quotient. It is assumed that the binary points of the dividend and divisor are originally lined up before they are loaded into this divide circuit. The actual position of the binary point is not important as long as it is in the same position for both operands.

The quotient has a binary point in the middle of the 16 bits, thus it always consists of an 8-bit integer portion and an 8-bit fractional portion. The circuit, as shown, may be used at clock frequencies up to 5 MHz. At this speed, the result is generated in 2.2 to 5 μs (11 to 25 clock periods). If the divisor contains a ONE in the most significant position the minimum divide time (2.2 μs) results. If the divisor is all ZEROs except for a ONE in the least significant position, the maximum divide time (5 μs) results. If speed is critical, the maximum time for a division may be decreased to about 2 μs with 9340s, 9341s, or 93S41s in place of the 9383s, using high speed versions of some of the other devices, and increasing the clock frequency.

This circuit starts its operation by first shifting the divisor and the dividend (if necessary) until they are of a proper magnitude for dividing. It then performs a series of shift and conditional subtract operations, the results of which determine the contents of the quotient register. X and Y are stored in two shift registers (9300s). The parallel inputs to the X register are multiplexed from two sources, either the dividend inputs (when loading) or the outputs of the subtractor (9383s), supplying the remainder of a subtraction, multiplied by two, through a 1-bit shift to the right.

When performing a division, the circuit cycles through the following steps.

- The X and Y registers (9300s) are loaded. The counter (9316) and quotient register (9334s) are reset. The two flip-flops are set.
- Both the X register and the Y register are shifted until either X_7 or Y_7 is a ONE (High). This does not affect the placement of the binary point in the result.
- A priority encoder (9318) is used to load the counter (9316) with a number corresponding to the placement of the binary point in the result. (If both X_7 and Y_7 are ONE, a seven is loaded into the counter. If Y contains leading ZEROs, a correspondingly smaller number is loaded). The Control Flip-Flop (CFF) is reset.
- If Y_7 is still ZERO, the Y register is shifted until Y_7 is a ONE, but the counter is not allowed to count. Dividend and divisor are now properly aligned for the division to start.
- Division is started by subtracting Y from X. If $X < Y$, X is shifted right ($\overline{PE}_x$ = High). If $X \geqslant Y$, X is loaded with the result of the subtraction, shifted one place ($\overline{PE}_x$ = Low), and a ONE is loaded into the latch (9334) position addressed by the counter. In either case, the counter is incremented. This step repeats until the counter reaches 15 (TC = High).
- CFF is set, division is complete.

Subtraction is not necessarily possible when X_7 is ONE. To permit the shifting of X in this case, an additional flip-flop ($\overline{X}_8$) is used to store the most significant bit. Subtraction is performed by adding the ones complement of Y to X and adding ONE by forcing a carry into the least significant adder stage. The most significant adder output is not used, since a successful subtraction always generates a ZERO in this position.

Any attempt to divide by zero is indicated by an output of the priority encoder. The "Division Complete" output is not activated.

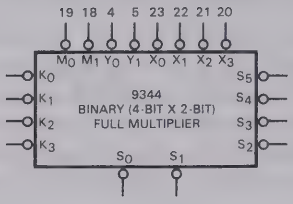
One precaution should be observed when using the circuit as shown. The $\overline{\text{Load}}$ input (normally one clock period wide) must persist beyond the falling edge of $\overline{\text{CP}}$, since this $\overline{\text{Load}}$ input is used as an asynchronous reset or set signal for the latches, counter and flip-flops, and they must be prevented from clocking on this clock edge.

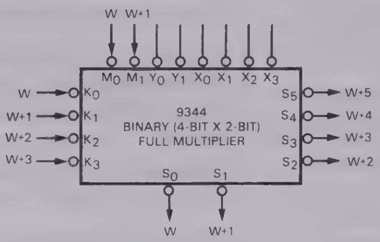
This divider does not attempt to round off the result, but provides an exact result in the form of a quotient and a remainder. The remainder may be re-entered and used to generate additional less significant bits. Alternatively, when shifting the dividend, instead of entering all ZEROs, a ONE followed by ZEROs or a ZERO followed by ONEs may be entered in the X registers. This has the same effect as generating a longer quotient and rounding it off, but the remainder is meaningless.

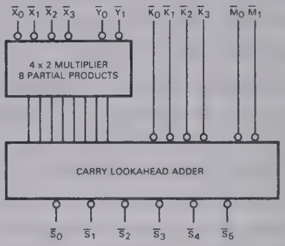
The 8-bit divider can easily be modified in one of several ways to meet specific system requirements. Expansion to 16 bits requires duplicating all MSI devices except the counter (an additional flip-flop suffices) and adding a few gates. If desired, shift registers (93164s) may be used in place of the addressable latches. In a serial system, the X and Y registers can be loaded serially and the quotient shifted out serially as it is generated.

The divider circuit can be modified to divide two floating point numbers, (exponential notation). Since, in this notation, the most significant bit of both the dividend and divisor is always a ONE, the 9318 priority encoder is not needed and the counter should always be loaded with the number seven. The division can be started immediately after loading the registers. A subtractor is needed to generate the exponent of the quotient by subtracting the exponent of the divisor from the exponent of the dividend. Assuming a number representation consisting of a fractional mantissa ($1 > X, Y \geqslant 1/2$) and an exponent, the quotient will be ($2 > Z \geqslant 1/2$). The quotient register need only be nine bits long, but the ability to perform a 1-bit shift on the quotient and to add ONE to the exponent must be included in the circuitry.

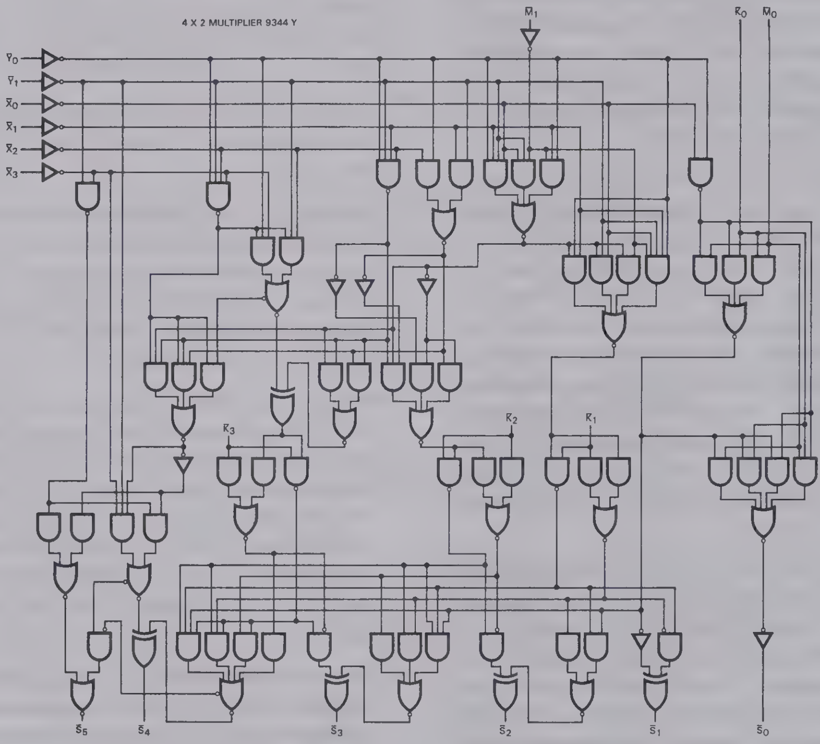
9344 4 BY 2-BIT BINARY MULTIPLIER DESCRIPTION


19 18 4 5 23 22 21 20
M_0 M_1 Y_0 Y_1 X_0 X_1 X_2 X_3
K_0
K_1
K_2
K_3
9344
BINARY (4-BIT X 2-BIT)
FULL MULTIPLIER
S_5
S_4
S_3
S_2
S_0 S_1


W W+1
M_0 M_1 Y_0 Y_1 X_0 X_1 X_2 X_3
W
W+1
W+2
W+3
K_0
K_1
K_2
K_3
9344
BINARY (4-BIT X 2-BIT)
FULL MULTIPLIER
S_5
S_4
S_3
S_2
W+5
W+4
W+3
W+2
S_0 S_1
W W+1


$\overline{X}_0$ $\overline{X}_1$ $\overline{X}_2$ $\overline{X}_3$ $\overline{Y}_0$ $\overline{Y}_1$ $\overline{K}_0$ $\overline{K}_1$ $\overline{K}_2$ $\overline{K}_3$ $\overline{M}_0$ $\overline{M}_1$
4 x 2 MULTIPLIER
8 PARTIAL PRODUCTS
CARRY LOOKAHEAD ADDER
$\overline{S}_0$ $\overline{S}_1$ $\overline{S}_2$ $\overline{S}_3$ $\overline{S}_4$ $\overline{S}_5$

This block represents the basic 4 by 2-bit multiplier, and indicates the weighting factors (power of two) attached to each of the inputs and outputs.


4 X 2 MULTIPLIER 9344 Y
$\overline{M}_1$
$\overline{K}_0$ $\overline{M}_0$
$\overline{Y}_0$
$\overline{Y}_1$
$\overline{X}_0$
$\overline{X}_1$
$\overline{X}_2$
$\overline{X}_3$
$\overline{K}_3$
$\overline{K}_2$
$\overline{K}_1$
$\overline{S}_5$
$\overline{S}_4$
$\overline{S}_3$
$\overline{S}_2$
$\overline{S}_1$
$\overline{S}_0$

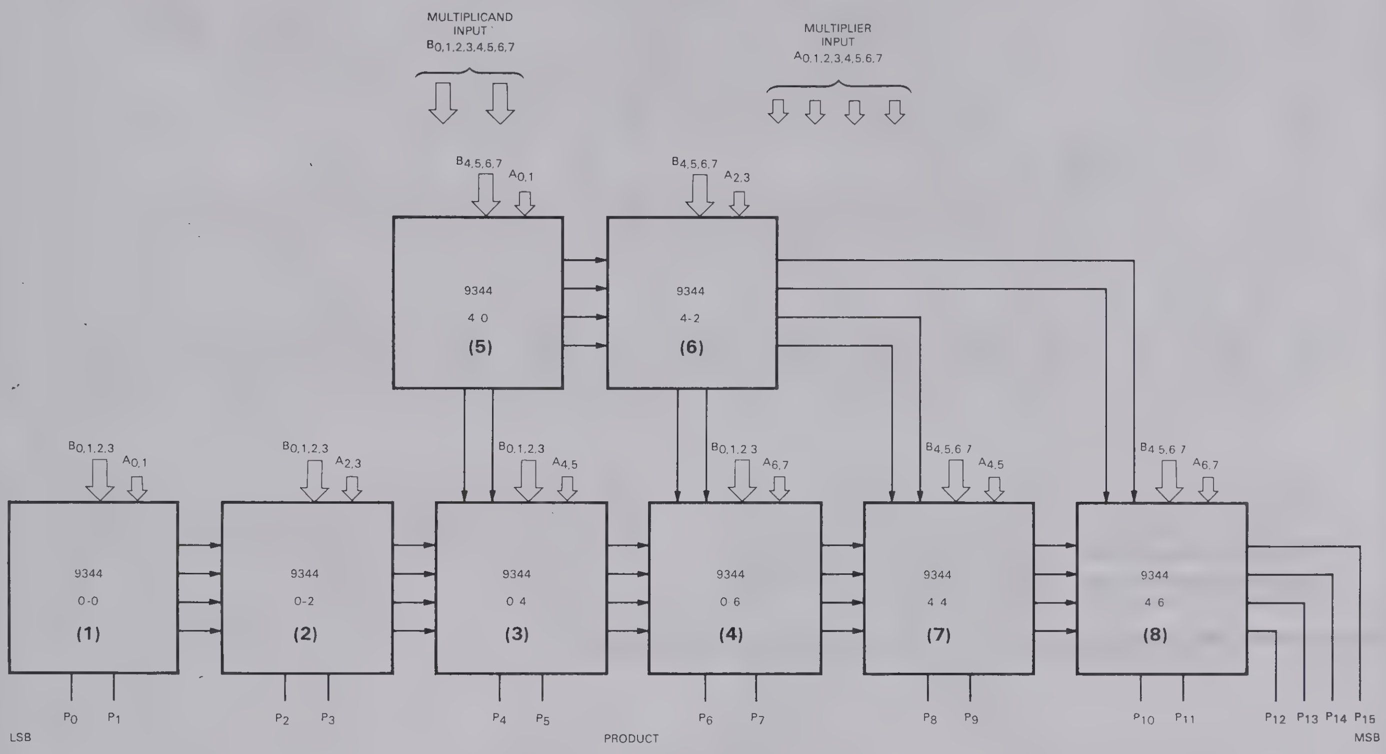
Multiplication is one of the more common and time consuming operations in data processing. Most computers multiply by a series of additions and shifts, a process that is acceptable for occasional multiplication (as in business computations) but is very time consuming in many scientific calculations (matrix inversions, Fast Fourier Transform) and becomes intolerably slow when these calculations are performed in a real time system.

A faster method is direct, combinatorial multiplication with a circuit that accepts two factors and generates the product without any sequential delays. Such a circuit uses a gate array to multiply (AND) each bit of factor A with each bit of factor B, and an adder array to add all partial products by binary weight. A 16 by 16-bit multiplication generates a 256-bit partial product for a 32-bit sum. This addition is fairly involved and must be performed with several levels of adders, because many partial products have identical weights.

The 9344 binary full multiplier offers a simple, fast and economical solution. It can be used as a building block (iterative cell) to build high speed binary multipliers of any size, without requiring any additional components.

The 9344 multiplies one set (X) of four inputs with another set (Y) of two inputs, generates eight partial products and adds them in a fast carry lookahead adder. It also adds two sets of additive inputs, one four bits wide (K), the other two bits wide (M), generating a 6-bit sum at the S outputs. The range for X and K is: 0 – 15, the range for Y and M is 0 – 3. The range for the sum output is 0 – (3 x 15 + 15 + 3) = 63.

8-BIT BY 8-BIT MULTIPLICATION ARRAY


MULTIPLICAND INPUT $B_{0,1,2,3,4,5,6,7}$
MULTIPLIER INPUT $A_{0,1,2,3,4,5,6,7}$
$B_{4,5,6,7}$ $A_{0,1}$
$B_{4,5,6,7}$ $A_{2,3}$
9344 4 0 (5)
9344 4-2 (6)
$B_{0,1,2,3}$ $A_{0,1}$
$B_{0,1,2,3}$ $A_{2,3}$
$B_{0,1,2,3}$ $A_{4,5}$
$B_{0,1,2,3}$ $A_{6,7}$
$B_{4,5,6,7}$ $A_{4,5}$
$B_{4,5,6,7}$ $A_{6,7}$
9344 0-0 (1)
9344 0-2 (2)
9344 0 4 (3)
9344 0 6 (4)
9344 4 4 (7)
9344 4 6 (8)
P_0 P_1 P_2 P_3 P_4 P_5 P_6 P_7 P_8 P_9 P_{10} P_{11} P_{12} P_{13} P_{14} P_{15}
LSB
PRODUCT
MSB

Binary Weight of Inputs and Outputs on the 9344s

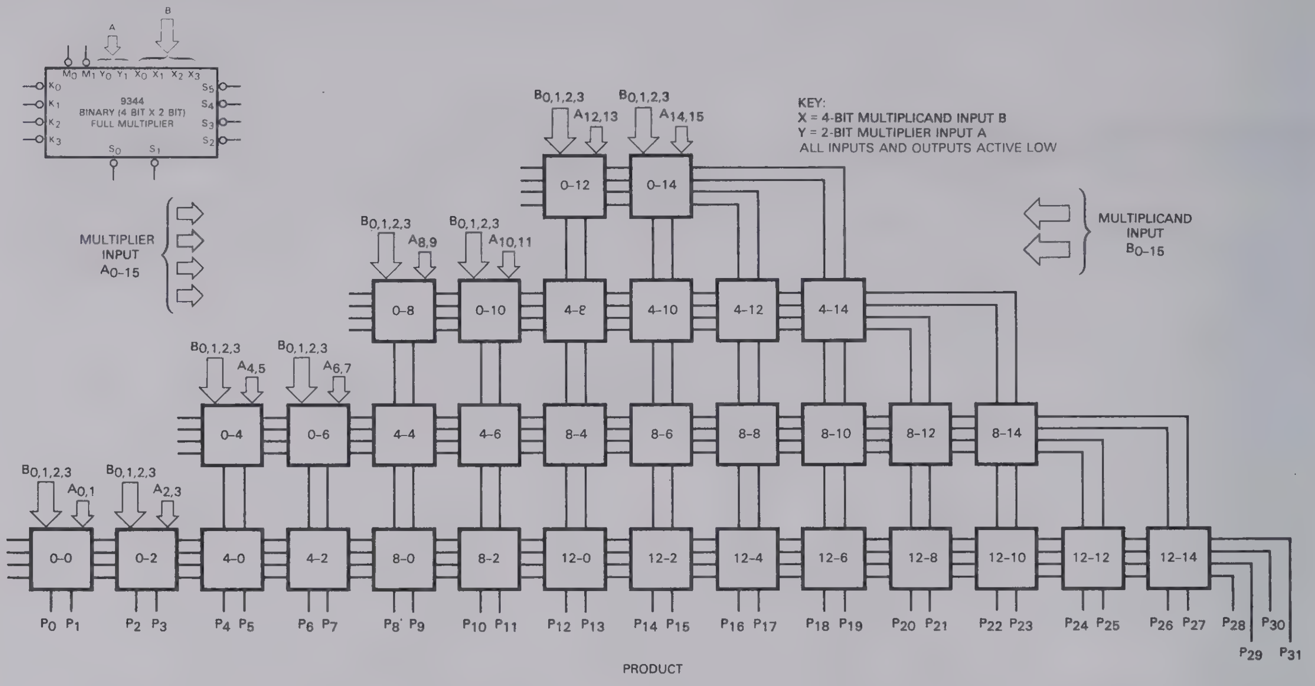
	X	Y	S	M	K
(1)	0,1,2,3	0,1	0,1,2,3,4,5	0,1	0,1,2,3
(2)	0,1,2,3	2,3	2,3,4,5,6,7	2,3	2,3,4,5
(3)	0,1,2,3	4,5	4,5,6,7,8,9	4,5	4,5,6,7
(4)	0,1,2,3	6,7	6,7,8.9,10,11	6,7	6,7,8.9
(5)	4,5,6,7	0,1	4,5,6,7,8,9	4,5	4,5,6,7
(6)	4,5,6,7	2,3	6,7,8,9,10,11	6,7	6,7,8.9
(7)	4,5,6,7	4,5	8,9,10,11,12,13	8,9	8,9,10,11
(8)	4,5,6,7	6,7	10,11,12,13,14,15	10,11	10,11,12,13

The 9344 can be used to build binary multipliers of any size, using one device for every eight partial products. For example, an 8 by 8-bit multiplier generating 64 partial products requires eight 9344s. The first 9344 multiplies the lowest four bits of B with the lowest two bits of A, the second 9344 multiplies the lowest four bits of B with the next two bits of A, etc.

The table shows the binary weight (power of two) associated with each input and output of the eight devices in an 8 by 8-bit multiplier. It indicates that many S outputs have the same weight and must, therefore, be added together. Since the 9344 has six additive inputs, these can be used to reduce the outputs to one set of S_{0-15} outputs, representing the 16-bit result.

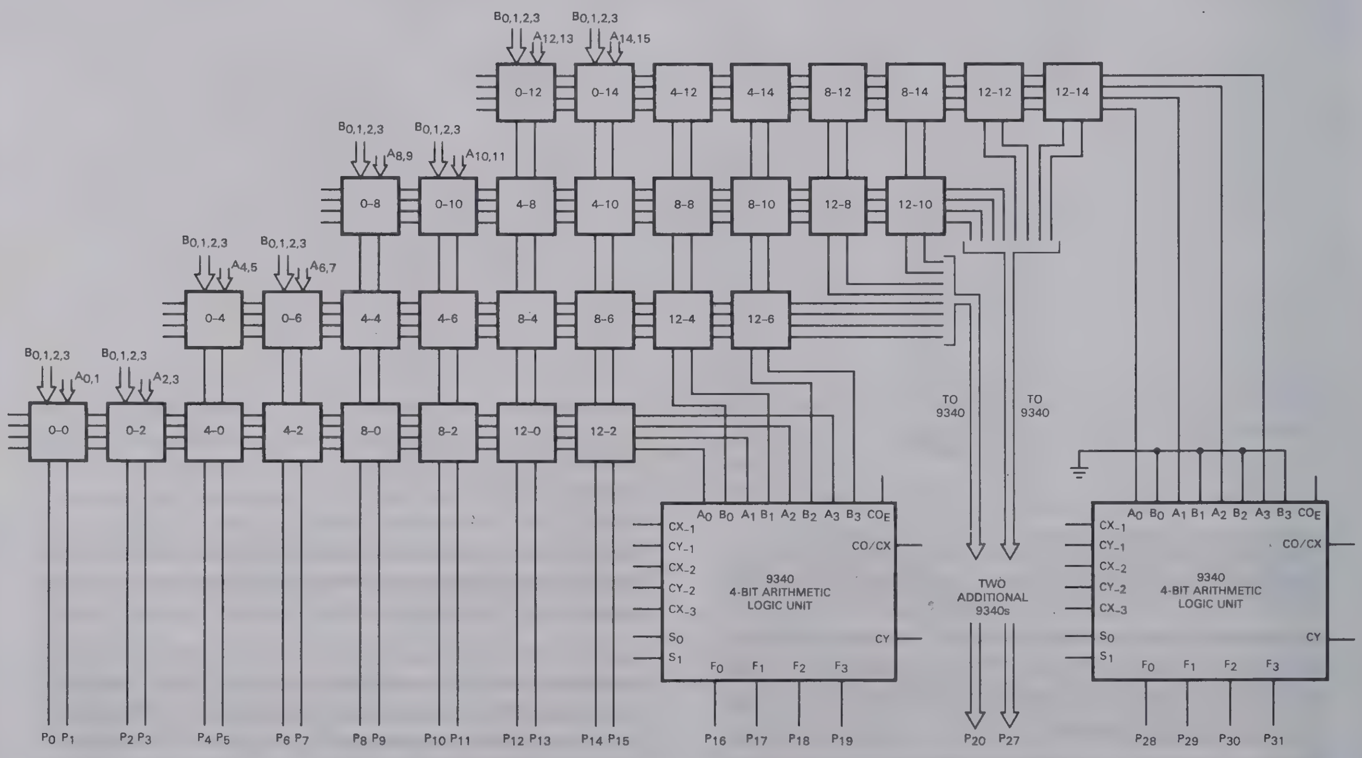
Speed: The 9344 uses internal carry lookahead, but propagation between devices is by rippling. The delay from M to the outputs S is typically 40 ns; the delay from inputs K to the outputs S is typically 25 ns. With proper interconnection only the K-to-output delays form a critical path. This 8 by 8-bit multiplier has a delay of 6 x 25 ns = 150 ns (typ).

16-BIT BY 16-BIT MULTIPLICATION ARRAY


A
B
M0 M1 Y0 Y1 X0 X1 X2 X3
K0
K1
K2
K3
S5
S4
S3
S2
S0 S1
9344
BINARY (4 BIT X 2 BIT)
FULL MULTIPLIER
KEY:
X = 4-BIT MULTIPLICAND INPUT B
Y = 2-BIT MULTIPLIER INPUT A
ALL INPUTS AND OUTPUTS ACTIVE LOW
MULTIPLIER INPUT A0–15
MULTIPLICAND INPUT B0–15
B0,1,2,3 A0,1
B0,1,2,3 A2,3
B0,1,2,3 A4,5
B0,1,2,3 A6,7
B0,1,2,3 A8,9
B0,1,2,3 A10,11
B0,1,2,3 A12,13
B0,1,2,3 A14,15
0–12
0–14
0–8
0–10
4–8
4–10
4–12
4–14
0–4
0–6
4–4
4–6
8–4
8–6
8–8
8–10
8–12
8–14
0–0
0–2
4–0
4–2
8–0
8–2
12–0
12–2
12–4
12–6
12–8
12–10
12–12
12–14
P0 P1 P2 P3 P4 P5 P6 P7 P8 P9 P10 P11 P12 P13 P14 P15 P16 P17 P18 P19 P20 P21 P22 P23 P24 P25 P26 P27 P28 P29 P30 P31
PRODUCT

In this 16 by 16-bit multiplier, the critical path involves 14 K-to-S output delays or 350 ns (typical). Each block represents one 9344. Labels inside the blocks identify bits multiplied in that block. The first number is the 4-bit B input, and the second number is the 2-bit A input. For instance, 12–0 refers multiplicand bits $B_{12,13,14,15}$ and multiplier bits $A_{0,1}$.

INCREASING THE SPEED OF A 16 BY 16-BIT BINARY MULTIPLIER

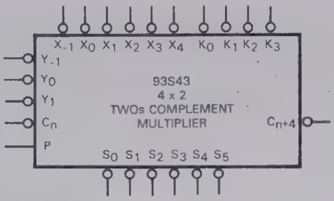

B0,1,2,3 A0,1
B0,1,2,3 A2,3
B0,1,2,3 A4,5
B0,1,2,3 A6,7
B0,1,2,3 A8,9
B0,1,2,3 A10,11
B0,1,2,3 A12,13
B0,1,2,3 A14,15
0–12
0–14
4–12
4–14
8–12
8–14
12–12
12–14
0–8
0–10
4–8
4–10
8–8
8–10
12–8
12–10
0–4
0–6
4–4
4–6
8–4
8–6
12–4
12–6
0–0
0–2
4–0
4–2
8–0
8–2
12–0
12–2
TO 9340
TO 9340
A0 B0 A1 B1 A2 B2 A3 B3 COE
CX-1
CY-1
CX-2
CY-2
CX-3
S0
S1
CO/CX
CY
F0 F1 F2 F3
9340
4-BIT ARITHMETIC
LOGIC UNIT
TWO ADDITIONAL 9340s
P0 P1 P2 P3 P4 P5 P6 P7 P8 P9 P10 P11 P12 P13 P14 P15 P16 P17 P18 P19 P20 P27 P28 P29 P30 P31

The 16 by 16-bit multiplier just described can be made even faster by rearranging the structure and performing some of the additions in external fast adders (9340s). This decreases the delay to 250 ns (typ).

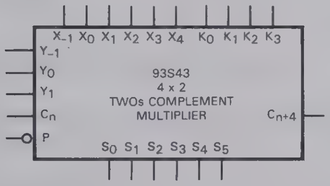
MULTIPLIERS/DIVIDERS

93S43 4-BIT BY 2-BIT TWOs COMPLEMENT MULTIPLIER DESCRIPTION AND OPERATION

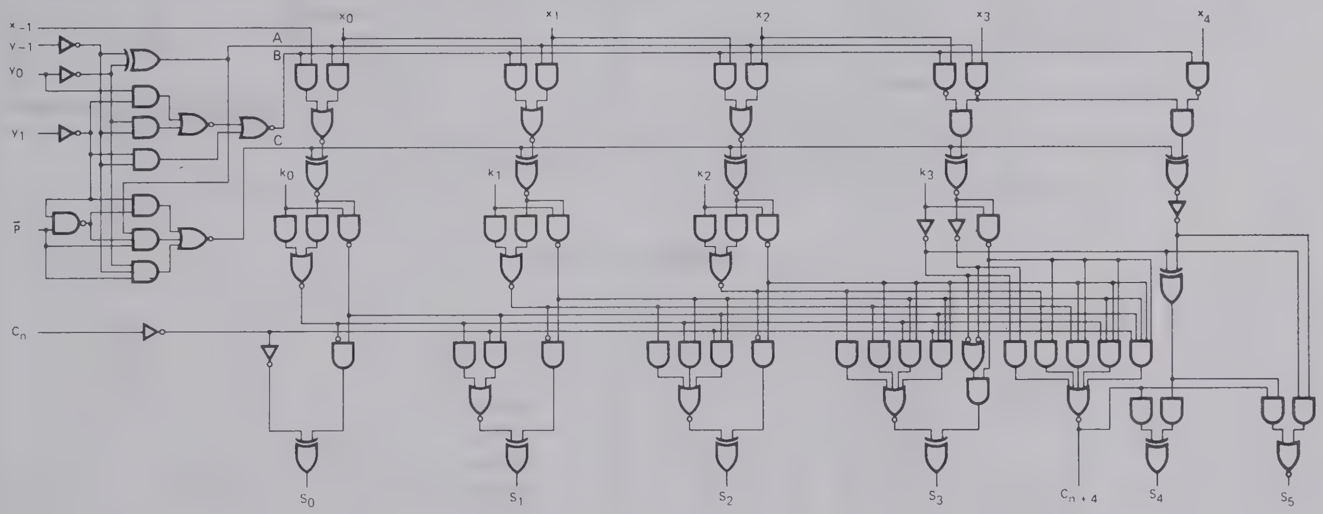
ACTIVE LOW


X_{-1} X_0 X_1 X_2 X_3 X_4 K_0 K_1 K_2 K_3
Y_{-1}
Y_0
Y_1
C_n
P
93S43
4 x 2
TWOs COMPLEMENT
MULTIPLIER
C_{n+4}
S_0 S_1 S_2 S_3 S_4 S_5

ACTIVE HIGH


X_{-1} X_0 X_1 X_2 X_3 X_4 K_0 K_1 K_2 K_3
Y_{-1}
Y_0
Y_1
C_n
P
93S43
4 x 2
TWOs COMPLEMENT
MULTIPLIER
C_{n+4}
S_0 S_1 S_2 S_3 S_4 S_5

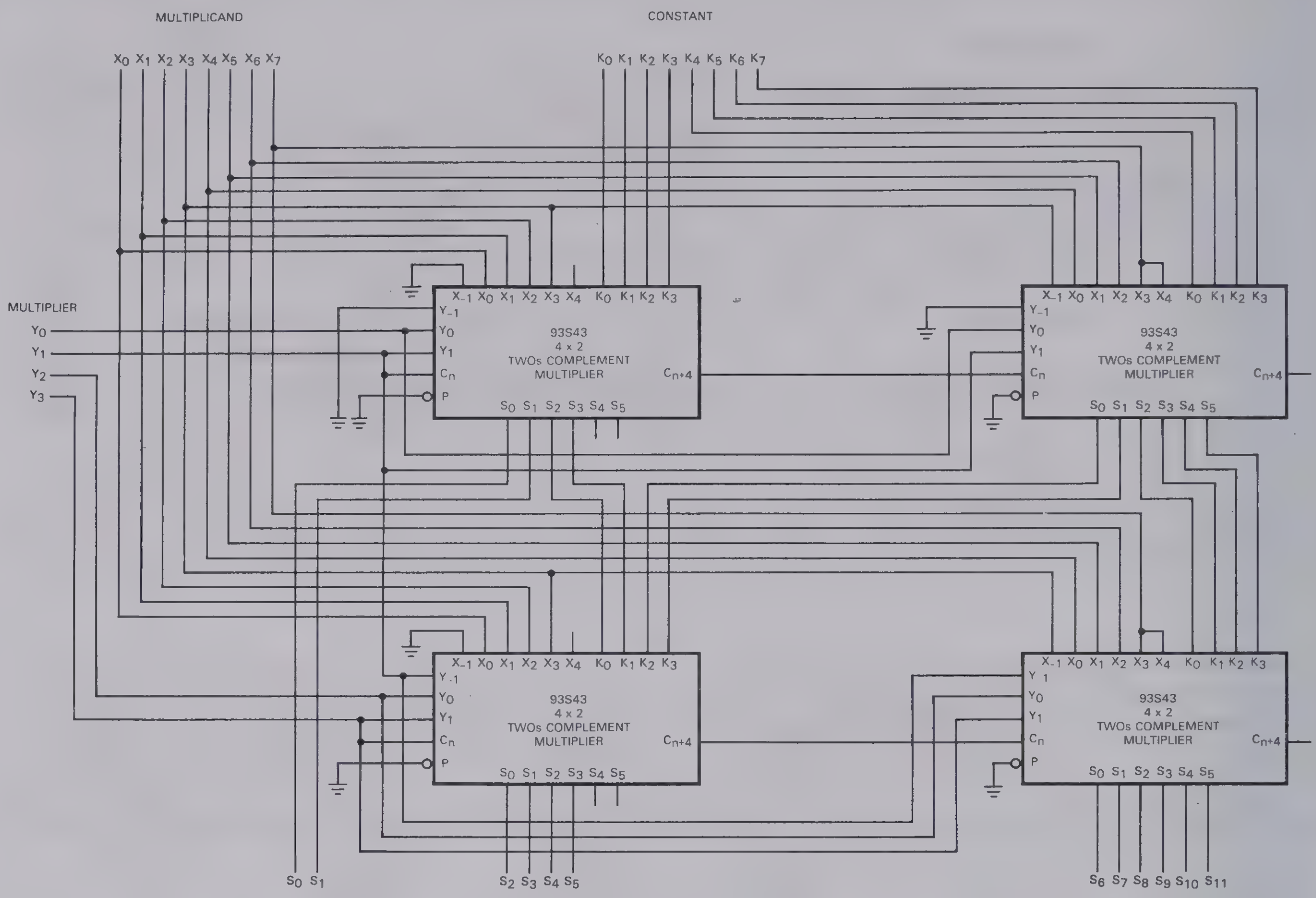
LEADS		LOADING
X_{-1}, X_3, X_4	Multiplicand Inputs	1 UL
X_0, X_1, X_2	Multiplicand Inputs	2 UL
Y_0	Multiplier Input	2 UL
Y_{-1}, Y_1	Multiplier Inputs	1 UL
K_0, K_1, K_2, K_3	Constant Inputs	2 UL
C_n	Carry Input	1 UL
P	Polarity Control (Active Low for High Operands)	3 UL
$S_0, S_1, S_2, S_3, S_4, S_5$	Product Outputs	12.5 UL
C_{n+4}	Carry Output	12.5 UL


x_{-1}
Y_{-1}
Y_0
Y_1
$\overline{P}$
C_n
A
B
C
x_0
x_1
x_2
x_3
x_4
k_0
k_1
k_2
k_3
S_0
S_1
S_2
S_3
C_{n+4}
S_4
S_5

The 93S43 is a super high speed digital multiplier building block which multiplies binary numbers represented in twos complement notation and generates a twos complement product directly, without corrections. The 93S43 is a 4 x 2-bit multiplier that can be connected to form iterative arrays multiplying binary numbers of any length, either directly or in a time sequenced arrangement. The device assumes that the most significant bit of the multiplier and the multiplicand has a negative weight. It can be used in arrays where the two factors have different word lengths. This multiplier uses the quaternary algorithm and performs the function S = (X times Y) plus K, where K is the input field used to add the partial products generated in the array. At the outside of the array the K inputs can be used to add a signed constant to the least significant part of the product.

The 93S43 operates with either active High or active Low inputs and outputs, provided the polarity control is properly terminated — grounded for active High operands (positive logic), terminated High for active Low operands (negative logic). The 93S43 is functionally and lead equivalent to the Am2505 and can be used as a plug-in replacement, but since the 93S43 uses Schottky technology, it is significantly faster. Through delay is improved by 40% (reduced to 60% of the values given for the Am2505) which allows either a 66% increase in system clock rate, or a simpler interconnection scheme at the same clock rate.

8 BY 4-BIT BINARY MULTIPLIER


MULTIPLICAND
X_0 X_1 X_2 X_3 X_4 X_5 X_6 X_7
CONSTANT
K_0 K_1 K_2 K_3 K_4 K_5 K_6 K_7
MULTIPLIER
Y_0
Y_1
Y_2
Y_3
X_{-1} X_0 X_1 X_2 X_3 X_4 K_0 K_1 K_2 K_3
Y_{-1}
Y_0
Y_1
C_n
P
93S43
4 x 2
TWOs COMPLEMENT
MULTIPLIER
C_{n+4}
S_0 S_1 S_2 S_3 S_4 S_5
S_0 S_1
S_2 S_3 S_4 S_5
S_6 S_7 S_8 S_9 S_{10} S_{11}

Typical Multiplication Times

Array Size	Number of 93S43s Used	Through Delay
4 x 4	2	40 ns
8 x 8	8	80 ns
12 x 12	18	125 ns
16 x 16	32	165 ns
24 x 24	72	150 ns

Multiplication time can be reduced by using additional carry lookahead adders (93S40).

The 93S43 can be used as a building block to form large binary multiplier arrays. The number of 93S43 devices required is given by

$$\frac{M \times N}{8}$$

where M and N are the word lengths of the multiplier and the multiplicand respectively. As shown above, four 93S43s can be connected to form a 4 x 8 array, multiplying an 8-bit binary multiplicand X (twos complement notation) with a 4-bit binary multiplier Y (also in twos complement notation), generating a product in twos complement notation.

The scheme is shown for active High operands (positive logic), but can easily be changed to active Low operands (negative logic) by holding the polarity control input P High.

The X-1 and Y-1 inputs at the least significant level must be held at the zero level. The S_4 and S_5 outputs are ignored except at the most significant edge of the array. The K inputs allow the accumulation of partial results as information passes through the array. On the first level the K inputs are not required for this purpose and can therefore be used to add a number to the least significant part of the product. This feature is very useful, since many arithmetic operations consist of a series of multiplications and additions. For multiplication of larger words the array can be extended in both the X and the Y dimensions.

COMPARATOR SYSTEMS

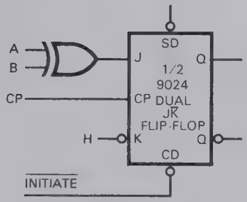
Comparator systems fall into two classes.

- Identity comparators, that detect whether or not two words are identical.
- Magnitude comparators, that also detect which of the two words is larger. Magnitude comparators are more complex and tend to be slower.

All comparators are defined in binary terms, but they can obviously be used with BCD or any other monotonic code without change.

IDENTITY COMPARATORS

BIT-SERIAL OPERATION

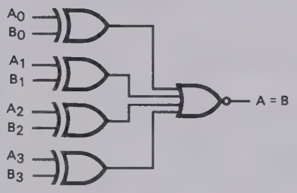

A
B
J
SD
Q
1/2
9024
DUAL
JK
FLIP-FLOP
CP
CP
H
K
CD
Q
INITIATE

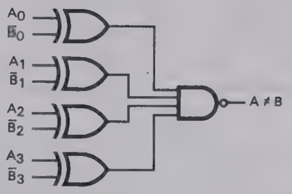
One Exclusive OR and one flip-flop form a serial identity comparator. The flip-flop must start out reset. As long as the A and B inputs are identical the output of the Exclusive OR is Low, leaving the flip-flop in its state. When $A \neq B$ the flip-flop is set and stays set until a new cycle is initiated by asynchronously clearing the flip-flop. The state of Q after the last bit has been clocked indicates the result of the comparison:

$Q : A \neq B$ $\overline{Q} : A = B$

Obviously the bit sequence does not affect the identity comparison.

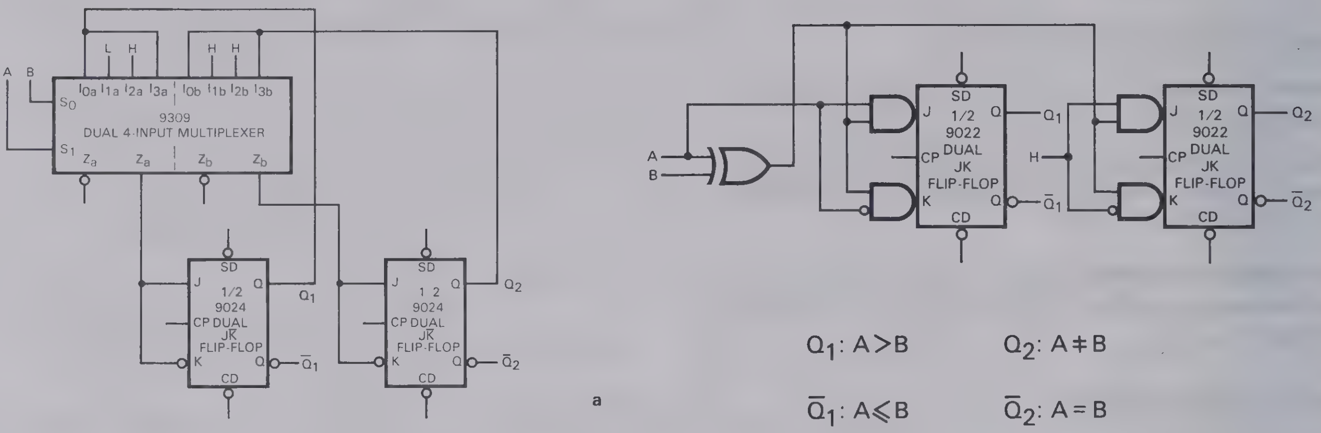
PARALLEL OPERATION


A_0
B_0
A_1
B_1
A_2
B_2
A_3
B_3
A = B


A_0
$\overline{B}_0$
A_1
$\overline{B}_1$
A_2
$\overline{B}_2$
A_3
$\overline{B}_3$
$A \neq B$

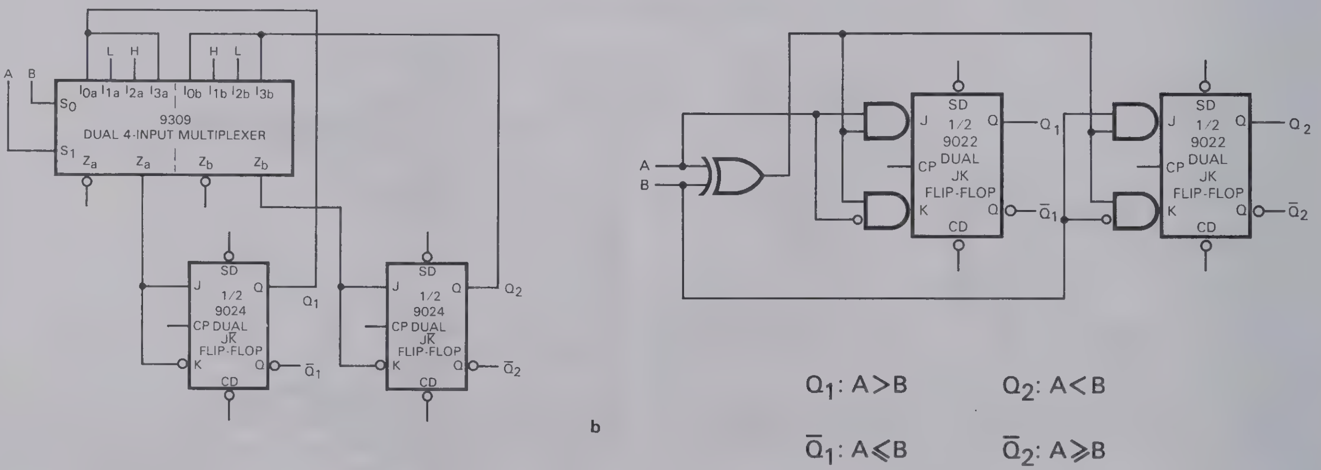
Parallel identity comparison is most efficiently performed with Quad Exclusive OR gates with outputs NORed or NANDed. The NAND configuration is faster, but requires opposite polarities of the two operands.

BIT SERIAL MAGNITUDE COMPARISON LEAST SIGNIFICANT BIT FIRST


A
B
L H
H H
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
S_1
9309
DUAL 4-INPUT MULTIPLEXER
Z_a Z_a Z_b Z_b
SD
J
Q
CP
K
CD
1/2
9024
DUAL
JK
FLIP-FLOP
Q_1
$\overline{Q}_1$
Q_2
$\overline{Q}_2$
H
9022
a

Q_1: A>B Q_2: A≠B

$\overline{Q}_1$: $A \leqslant B$ $\overline{Q}_2$: A = B


A
B
L H
H L
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
S_0
S_1
9309
DUAL 4-INPUT MULTIPLEXER
Z_a Z_a Z_b Z_b
SD
J
Q
CP
K
CD
1/2
9024
DUAL
JK
FLIP-FLOP
Q_1
$\overline{Q}_1$
Q_2
$\overline{Q}_2$
9022
b

Q_1: A>B Q_2: A<B

$\overline{Q}_1$: $A \leqslant B$ $\overline{Q}_2$: $A \geqslant B$

Magnitude comparison discriminates between three possible conditions. A > B, A < B, and A = B, usually encoded on two output signals.

A serial magnitude comparator for LSB first *(a)* is most efficiently implemented by either a dual 4-input multiplexer (9309) and a dual flip-flop (9024), or by an Exclusive OR gate and a dual flip-flop with Enable (9022). (Note that the 9022 master/slave flip-flop requires stable inputs during the entire clock Low period.)

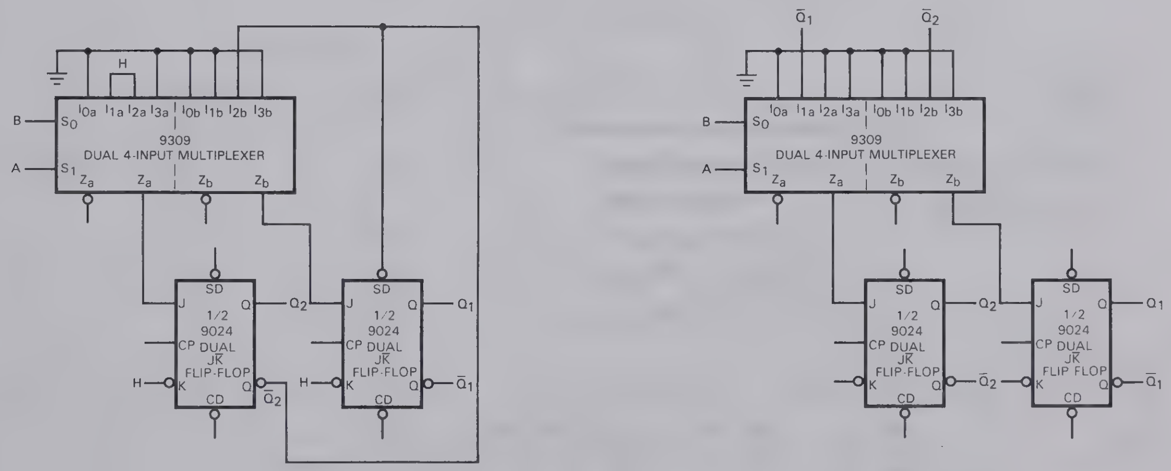
Assuming active High notation,

Q_1 is set by $A \cdot \overline{B}$, reset by $\overline{A} \cdot B$, unaffected by $A \cdot B$ or $\overline{A} \cdot \overline{B}$ (A = B)

Q_2 is set by A ≠ B, unaffected by A = B.

Thus, if both flip-flops start out reset, their state after clocking in the most significant bit indicates the result of the comparison. A slight rearrangement of the same basic circuit *(b)*, generates a different set of outputs.

MOST SIGNIFICANT BIT FIRST MAGNITUDE COMPARISON


H
B
A
S_0
S_1
I_{0a} I_{1a} I_{2a} I_{3a} I_{0b} I_{1b} I_{2b} I_{3b}
9309
DUAL 4-INPUT MULTIPLEXER
Z_a $\bar{Z}_a$ Z_b $\bar{Z}_b$
SD
J
K
CP
CD
1/2
9024
DUAL
JK
FLIP-FLOP
Q_1
$\bar{Q}_1$
Q_2
$\bar{Q}_2$

Q_1: $A > B$ Q_2: $A \neq B$

$\overline{Q}_1$: $A \leq B$ $\overline{Q}_2$: $A = B$

a

Q_1: $A > B$ Q_2: $A < B$

$\overline{Q}_1$: $A \leq B$ $\overline{Q}_2$: $A \geq B$

b

Magnitude comparison is also possible when the serial words come in "backward", with their most significant bits first *(a)*. In this case the first bit where A differs from B determines the result. This circuit sets Q_1 when $A \cdot \overline{B} \cdot \overline{Q}_2$, i.e., if $A > B$ and all previous bits have been $A = B$, leaving Q_1 unaffected under all other conditions. It sets Q_2 if $A \neq B$, but does not reset it until a new comparison is initiated by clearing both flip-flops.

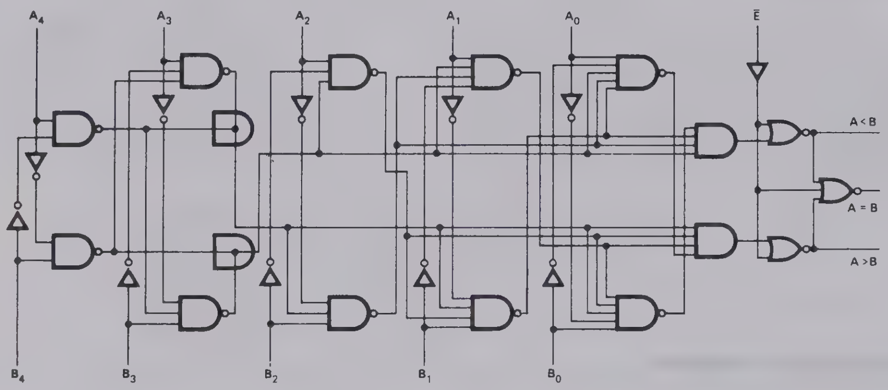
A slight rearrangement of basically the same circuit *(b)* generates a different set of outputs:

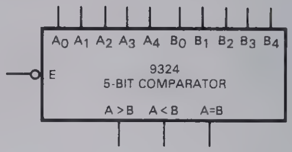
Q_1: $A > B$, Q_2: $A < B$.

COMPARATOR SYSTEMS

High speed parallel systems require a direct magnitude comparison over many bits in parallel. In a computer this function is usually performed by the arithmetic logic unit. Subtracting B from A yields a negative result if $A<B$, a positive result if $A>B$ and zero if $A = B$. If an isolated parallel comparison is needed, it is performed most economically by the 9324 5-bit magnitude comparator.

9324/93L24 5-BIT COMPARATOR





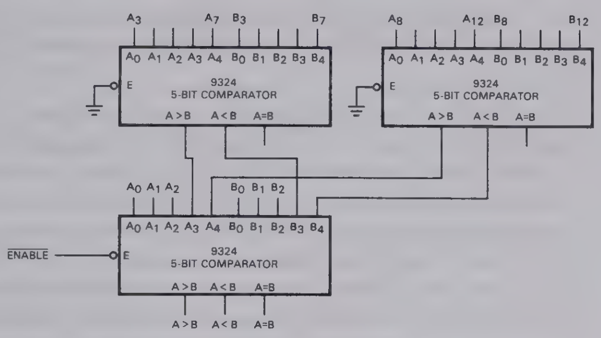
LEADS		LOADING	
		9324	93L24
$\overline{E}$	Enable (Active Low) Input	2 UL	.5
A_0, A_1, A_2, A_3, A_4	Word A Parallel Inputs	2 UL	.5
B_0, B_1, B_2, B_3, B_4	Word B Parallel Inputs	2 UL	.5
$A<B$	A Less than B Output	9 UL	2.25
$A>B$	A Greater than B Output	9 UL	2.25
$A = B$	A Equal to B Output	10 UL	2.50

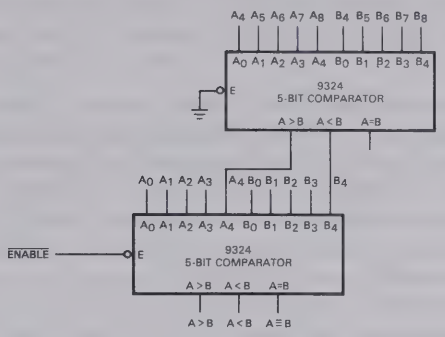
$\overline{E}$	Ay By	$A<B$	$A>B$	$A = B$
H	X X	L	L	L
L	Word A = Word B	L	L	H
L	Word A >Word B	L	H	L
L	Word A<Word B	H	L	L

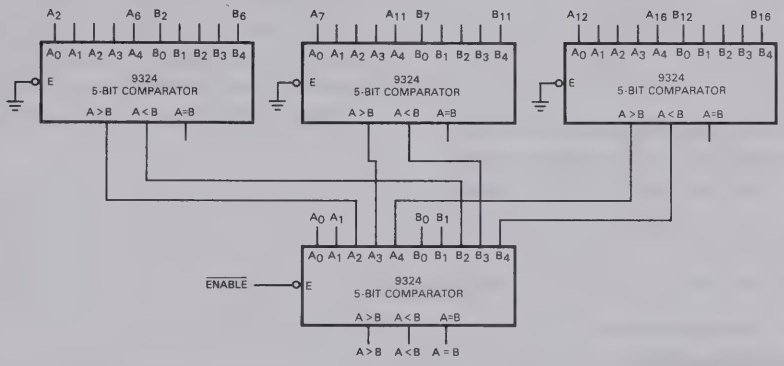
The 9324 is a 5-bit (or expandable 4-bit) magnitude comparator. It accepts two 5-bit numbers, A_{0-4} and B_{0-4} and generates three mutually exclusive, active High outputs: A greater than B, A less than B, and A equal to B. When the active Low Enable input is High, all outputs are forced Low. The delay from the operand inputs to the $A<B$ and $A>B$ outputs is a maximum of five gate delays, approximately 40 ns. The $A = B$ output is derived from the other two outputs and is therefore delayed by another gate. The 9324 might be ripple expanded as an expandable 4-bit comparator, but since it is a true 5-bit comparator, it can be expanded in parallel, resulting in much faster operation at no extra cost. Parallel comparator arrays are shown for up to 65 bits.

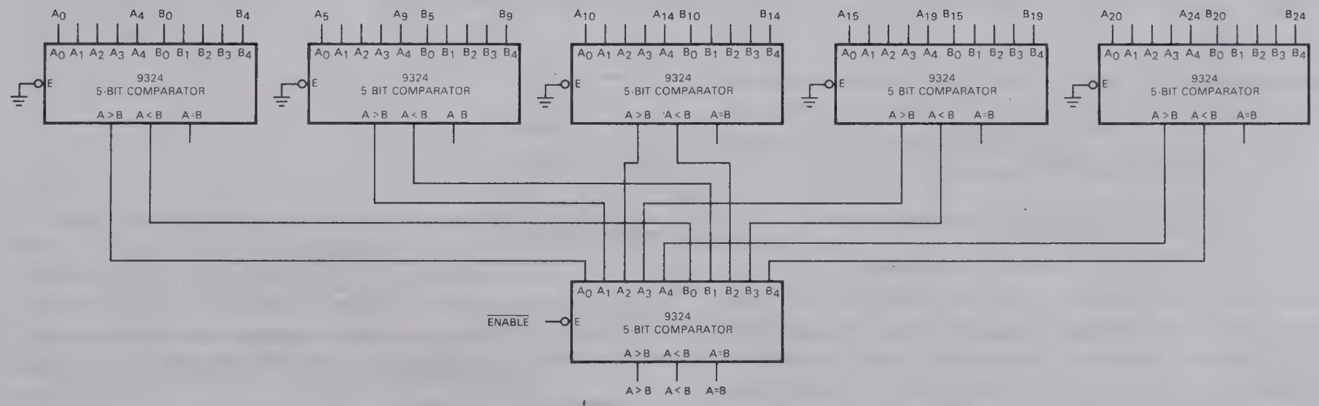
COMPARATOR SYSTEMS

PARALLEL COMPARATOR ARRAYS


10 TO 13 BITS
A3
A7
B3
B7
A8
A12
B8
B12
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
E
9324
5-BIT COMPARATOR
A > B
A < B
A=B
A0 A1 A2
B0 B1 B2
ENABLE

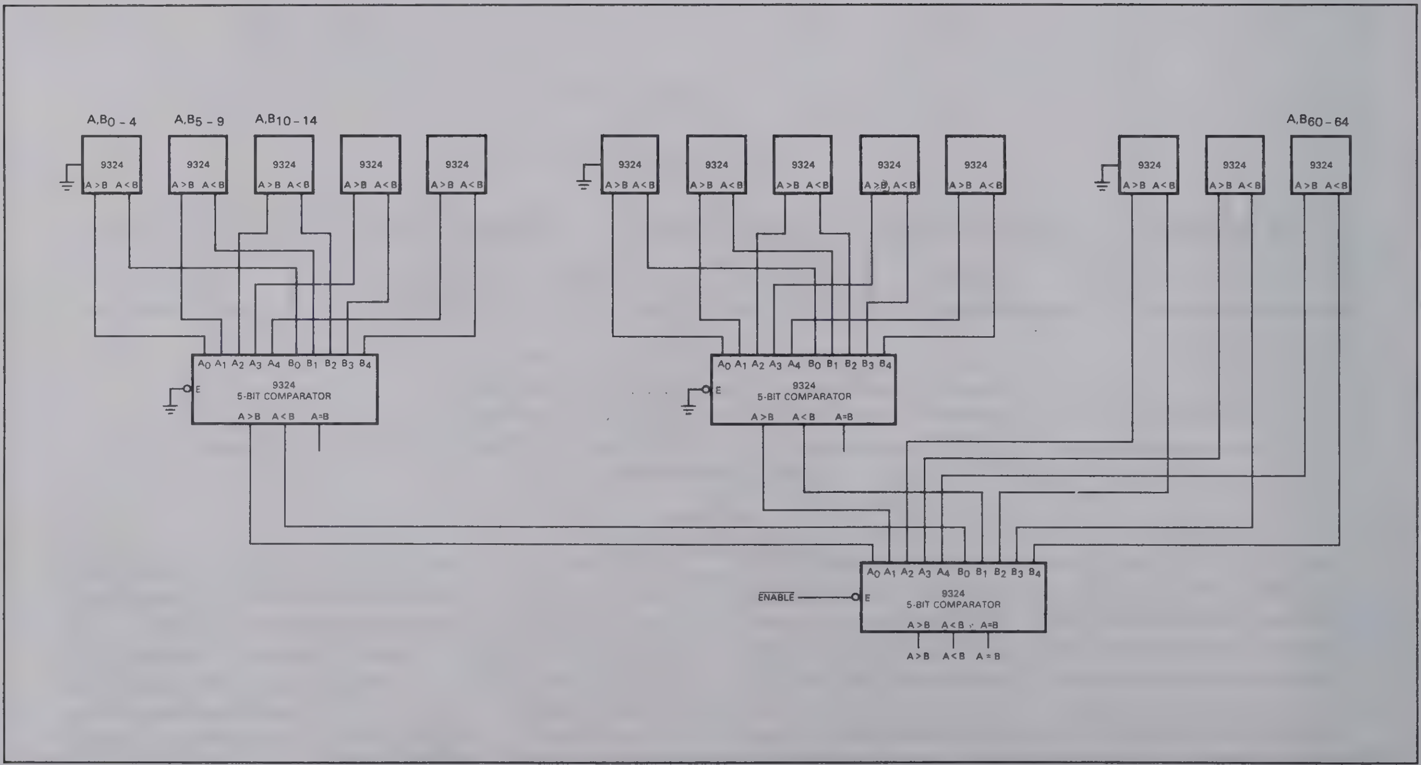

6 TO 9 BITS
A4 A5 A6 A7 A8 B4 B5 B6 B7 B8
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
E
9324
5-BIT COMPARATOR
A > B
A < B
A=B
A0 A1 A2 A3
A4 B0 B1 B2 B3
B4
ENABLE
A≡B


14 TO 17 BITS
A2
A6
B2
B6
A7
A11
B7
B11
A12
A16
B12
B16
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
E
9324
5-BIT COMPARATOR
A > B
A < B
A=B
A0 A1
B0 B1
ENABLE


22 TO 25 BITS
A0
A4
B0
B4
A5
A9
B5
B9
A10
A14
B10
B14
A15
A19
B15
B19
A20
A24
B20
B24
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
E
9324
5-BIT COMPARATOR
A > B
A < B
A=B
ENABLE

COMPARATOR SYSTEMS

65-BIT PARALLEL COMPARATOR ARRAY



A,B0 – 4
A,B5 – 9
A,B10 – 14
A,B60 – 64
9324
A>B A<B
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
E
9324
5-BIT COMPARATOR
A>B A<B A=B
ENABLE
A>B A<B A=B

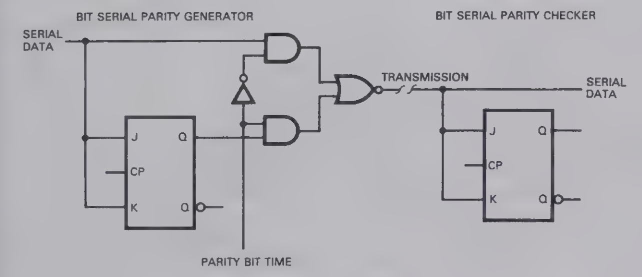
ERROR DETECTION/CORRECTION

Whenever digital data is transferred from one location to another, there is a probability for error due either to device failure or noise. There are numerous ways to handle errors at the system level. Some systems detect errors and request retransmission of data. In other systems, retransmission may be impossible or prohibitively expensive. In such systems, the receiving equipment must not only be able to detect, but also correct the error.

Both error detection and error correction rely on the transmission of redundant information. This requires additional bits of data and lowers the overall efficiency of transmission. In parallel systems additional wires, transmitters, and receivers are required, whereas serial transmission systems use additional time to transmit the redundant information. All these methods cannot completely eliminate errors, but as the percentage of redundant *data* bits increases, the probability of undetected or uncorrected errors decreases.

The simplest and most common method of dealing with errors is the addition of a single extra bit, called a *parity bit*. The parity bit is chosen such that the total number of ones in the word (counting the parity bit) is odd (in an odd parity system) or even (in an even parity system). Odd parity is generally preferred, since it insures at least one "1" in any word. At the receiving end, the parity of the word is examined. If any single bit in the word was changed, the detector indicates wrong parity. However, if an even number of errors occurs, this simple method cannot detect it. The parity bit provides only *single error detecting.*

PARITY GENERATION



In a serial parity generator, a flip-flop is toggled for every "1" in the data word and the state of this flip-flop is inserted as a trailing parity bit. On the receiving side, the parity checker has an equivalent flip-flop. Its state is interrogated after the data has been received. Both circuits are easily adapted for odd or even parity systems.

For parallel systems it is necessary to generate the modulo 2 sum of many inputs simultaneously. This requires an array of cascaded Exclusive OR circuits. The 9348 12-bit parity checker/generator is specifically designed for this function.

ERROR CORRECTION[1] — HAMMING CODES

A parity bit can only *detect* single errors. It cannot reliably detect multiple errors and it cannot *correct* single errors either. A single redundant bit does not carry enough information to do so. It is possible, however, to add more redundant information to the data, formulated such that errors are not only detected, but also corrected.

A data word containing an error-correcting field of redundant information is called a Hamming code. It uses a series of parity bits generated and arranged such that a unique set of parity errors results from an error in any given bit position. For example: Three redundancy bits can have a total of eight different states. Since one of these states must indicate "no error", the other seven states can be used to locate an error in any one of seven transmitted bits. Three of the transmitted bits are the redundancy bits themselves, leaving four data bits in which an error can be uniquely *detected*, and also *corrected*. The coding of the parity bits is done conveniently so the pattern of parity errors is the binary address of the bit in error. In general, a Hamming code contains $2^m - 1$ bits, m of which are the Hamming or check bits, $2^m - m - 1$ are the data bits.

Total Bits	Hamming Bits	Data Bits
7	3	4
15	4	11
31	5	26

Thus three additional parity (Hamming) bits can provide single error correction for 4-bit data words. The seven bits are arranged in the following way:

$$P_0\, P_1\, D_0\, P_2\, D_1\, D_2\, D_3$$

where D_0, D_1, D_2, D_3 are the four data bits

P_0 is odd parity over bits D_0, D_1, D_3

P_1 is odd parity over bits D_0, D_2, D_3

P_2 is odd parity over bits D_1, D_2, D_3

At the receiving end the three parity bits are again generated from the data bits using an identical scheme. Then these three parity bits are compared with the three transmitted parity bits. If they all match, there was no single error. If they differ, the pattern of mismatches is interpreted as a binary address of the bit in error.

A practical system would avoid the additional comparison and generates the error address (E_{0-2}) by including the received parity bits in the parity check:

E_0 is the odd parity over bits $P_0\, D_0\, D_1\, D_3$

E_1 is the odd parity over bits $P_1\, D_0\, D_2\, D_3$

E_2 is the odd parity over bits $P_2\, D_1\, D_2\, D_3$

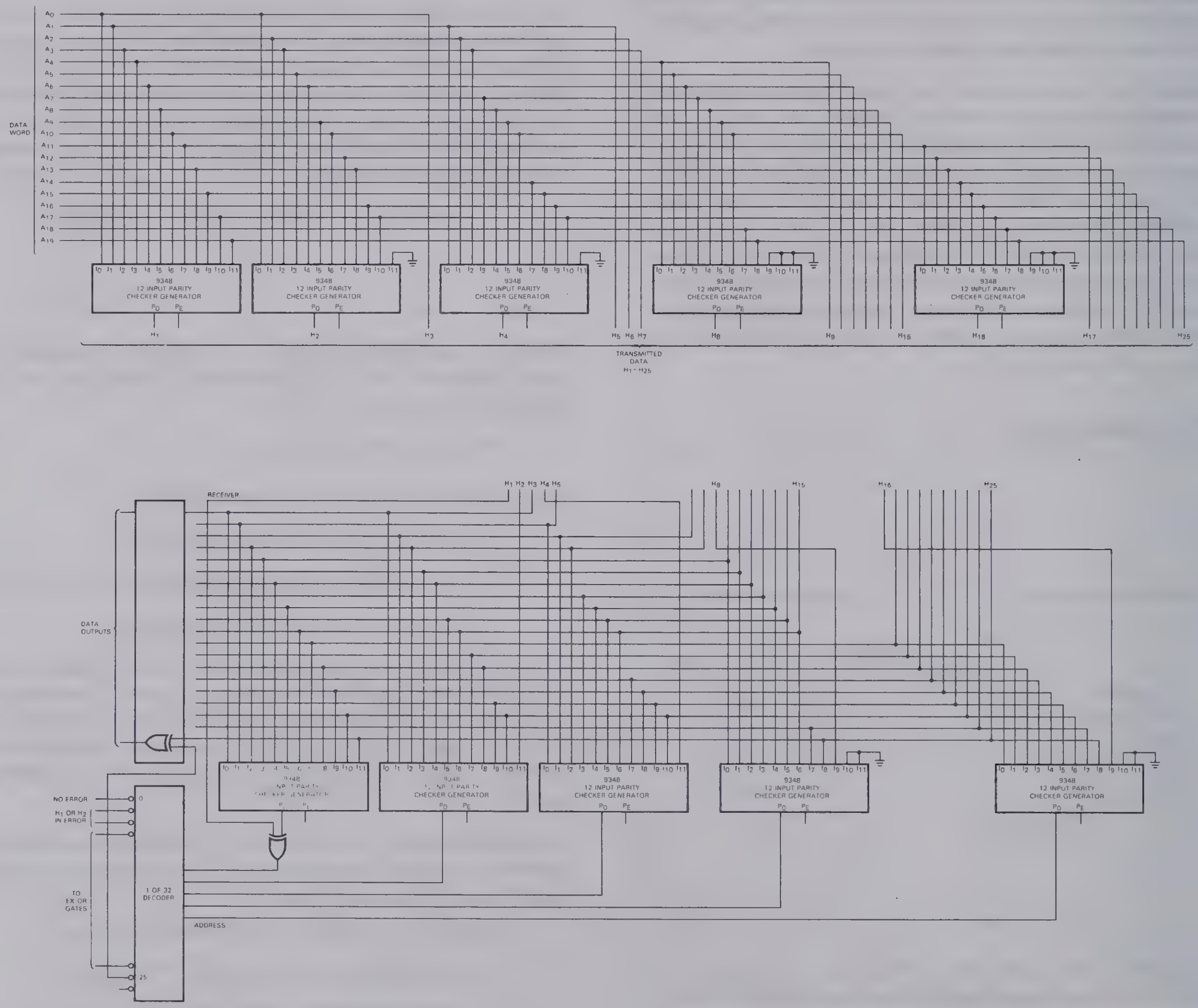
This Hamming code can detect and correct single errors, but it will fail on double errors. It would correct the wrong bit. If, however, one more overall parity bit is added, it is also possible to detect (but not correct) double errors. When the receiver finds the overall parity check correct and the error address is zero, there was no error. If the overall parity check is wrong and the error address is not zero, there was a single error which can be corrected. If, however, the overall parity check is correct, but the error address is not zero, then there was a non-correctable double error.

---

[1] For a detailed description of the theory behind and the applications of error correcting codes, see Peterson and Weldon, ERROR CORRECTING CODES, Second Edition, The MIT Press, Cambridge, Ma., 1972

ERROR DETECTION/CORRECTION

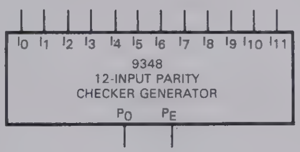
ERROR CORRECTING — HAMMING CODES


A0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
A15
A16
A17
A18
A19
DATA WORD
I0 I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11
9348
12 INPUT PARITY CHECKER GENERATOR
PO PE
H1
H2
H3
H4
H5 H6 H7
H8
H9
H15
H16
H17
H25
TRANSMITTED DATA H1 – H25
RECEIVER
H1 H2 H3 H4 H5
DATA OUTPUTS
NO ERROR
H1 OR H2 IN ERROR
TO EX-OR GATES
1 OF 32 DECODER
0
25
ADDRESS

This figure illustrates the Hamming circuit for a 20-bit data word. Five 9348s are used, and each bit of the data word is sent to several parity generators. As a result, information about the state of a given bit is contained in several bits of the 25-bit data plus parity Hamming code. The Hamming generator is duplicated at the receiving end, incorporating the received parity bits in the parity generation. This generates the 5-bit address for the bit in error. Address 00000 indicates "no error". This address can be fed to a 1-of-32 decoder, the outputs of which are used to invert the erroneous bit.

9348 12-INPUT PARITY CHECKER/GENERATOR

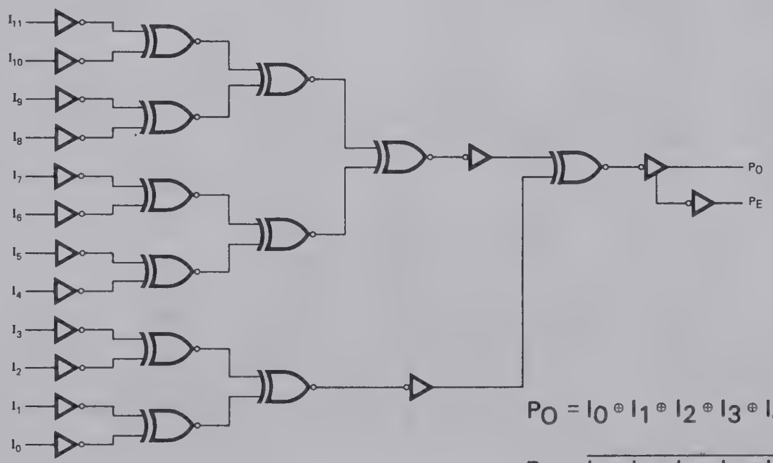
LEADS		LOADING
I_0 to I_{11}	Parity Inputs	2 UL
PO	Odd Parity Output	10 UL
PE	Even Parity Output	10 UL



INPUTS		OUTPUTS	
$I_0, I_1, I_2, I_3, I_4, I_5, I_6, I_7, I_8, I_9, I_{10}, I_{11}$		P_O	P_E
All Twelve	Inputs Low	L	H
Any One	Input High	H	L
Any Two	Inputs High	L	H
Any Three	Inputs High	H	L
Any Four	Inputs High	L	H
Any Five	Inputs High	H	L
Any Six	Inputs High	L	H
Any Seven	Inputs High	H	L
Any Eight	Inputs High	L	H
Any Nine	Inputs High	H	L
Any Ten	Inputs High	L	H
Any Eleven	Inputs High	H	L
Any Twelve	Inputs High	L	H

The 9348 is a 12-input parity checker/generator that produces both odd and even parity outputs. It is generally used for error detection applications, including the generating and checking of parity codes and error correcting Hamming codes. The 9348 can be used for any number of inputs up to 12, particularly for the popular 9-bit byte format, and it can also be easily cascaded for longer words.

9348 OPERATION



$$P_O = I_0 \oplus I_1 \oplus I_2 \oplus I_3 \oplus I_4 \oplus I_5 \oplus I_6 \oplus I_7 \oplus I_8 \oplus I_9 \oplus I_{10} \oplus I_{11}$$

$$P_E = \overline{I_0 \oplus I_1 \oplus I_2 \oplus I_3 \oplus I_4 \oplus I_5 \oplus I_6 \oplus I_7 \oplus I_8 \oplus I_9 \oplus I_{10} \oplus I_{11}}$$

The 9348 accepts 12 data inputs and provides both odd and even parity outputs. The Even Parity output (P_E) is High if an even number of inputs is active (High or Low) while the Odd Parity output (P_O) is High if an odd number of the 12 inputs is active (High or Low). The Even Parity output (P_E) is generated from the Odd Parity output (P_O) through an inverter. The two outputs are therefore always complementary and P_O is always the faster output. The logic diagram shows the internal logic design of the 9348, using a multilevel Exclusive NOR structure. The inverters are level restorers and cause a 1-gate delay. The through delay from the first four inputs to the outputs is one Exclusive NOR delay shorter than the through delay from the other eight inputs. This feature can be used to balance system delays by applying later signals to these faster inputs I_{0-3}.

OPTIMUM TERMINATION OF UNUSED INPUTS

Number of Inputs	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}
5	D_0	L	D_1	L	D_2	L	D_3	L	D_4	L	L	L
6	D_0	L	D_1	L	D_2	L	D_3	L	D_4	L	D_5	L
7	D_0	D_1	D_2	L	D_3	L	D_4	L	D_5	L	D_6	L
8	D_0	D_1	D_2	D_3	D_4	L	D_5	L	D_6	L	D_7	L
9	D_0	D_1	D_2	D_3	D_4	D_5	D_6	L	D_7	L	D_8	L
10	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	L	D_9	L
11	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	L
12	ALL INPUTS USED											

The parity function of the 9348 shows a symmetry (inherent to the Exclusive NOR function) which allows additional flexibility. Pairs of unused inputs may be tied Low or High without changing the logic function performed on the other inputs. When some of the 12 inputs are not used, the delay through the 9348 can be minimized by appropriate termination of the unused inputs. The delay through an Exclusive NOR gate is minimized if the unused input is held High. Consequently, delays through the 9348 can be minimized by terminating unused inputs in such a way that all unused inputs of the internal Exclusive NOR gates are held High. The table gives appropriate input terminations for different input word lengths.

CODE CONVERSION

Numbers can be represented in a large variety of codes. The binary code is the most natural, the simplest, and the one most commonly used in high speed computer systems. For convenience this code is often grouped in 3-bit groups and called an "octal code", but since these are just different ways of interpreting the binary code, all its features are retained.

Unfortunately, a different numbering system, based on the number 10, is in everyday use, and also, mixed numbering systems are used for some special applications (time, angles, etc.). This has created a need for binary-to-BCD (binary coded decimal) and BCD-to-binary converter circuits.

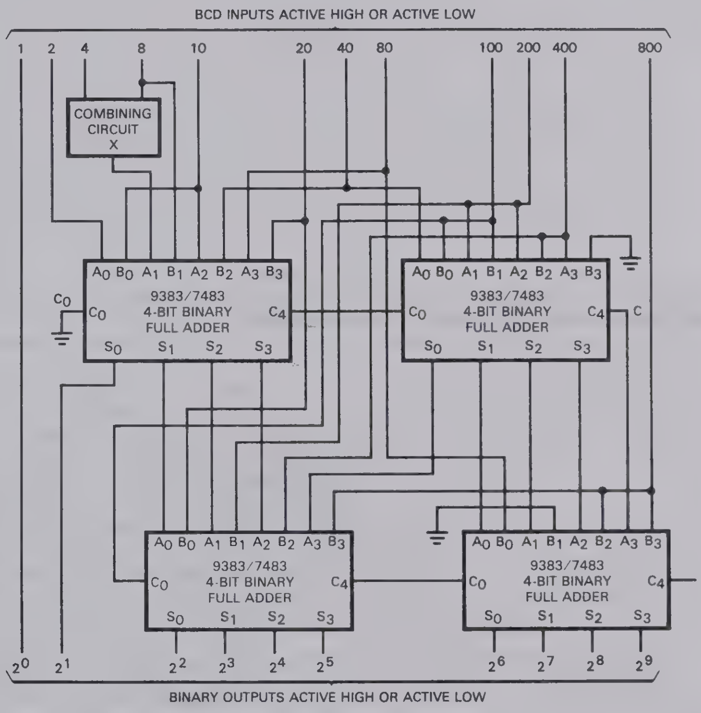
The number of bits and digits involved, the time available, and the amount of general purpose (perhaps even microprogrammed) logic available in the system are important factors in selecting one of the many different methods available for code conversion.

Any arbitrary code can be converted into any other arbitrary code by using a Read Only Memory (ROM) as a lookup table. This method is very fast with Bipolar ROMs, but in most cases it is unnecessarily expensive, since most codes show some kind of regularity. Cheaper and fewer MSI circuits can take advantage of this regularity and provide a more economical solution.

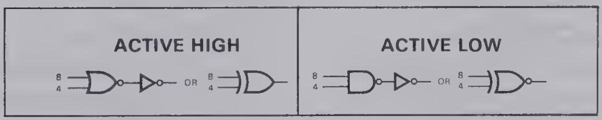
Binary adders are used in high speed parallel BCD-to-binary conversion. Every bit in a BCD number can be expressed as a binary number, and their sum is the binary equivalent of the whole BCD number.

BCD-TO-BINARY CONVERTER USING ADDERS

BCD Bits \ Binary Bits	1	2	4	8	16	32	64	128	256	512	1024	2048	4096	8192
1	X													
2		X												
4			X											
8				X										
10		X		X										
20			X		X									
40				X		X								
80					X		X							
100			X			X	X							
200				X			X	X						
400					X			X	X					
800						X			X	X				
1000				X		X	X	X	X	X				
2000					X		X	X	X	X	X			
4000						X		X	X	X	X	X		
8000							X		X	X	X	X	X	



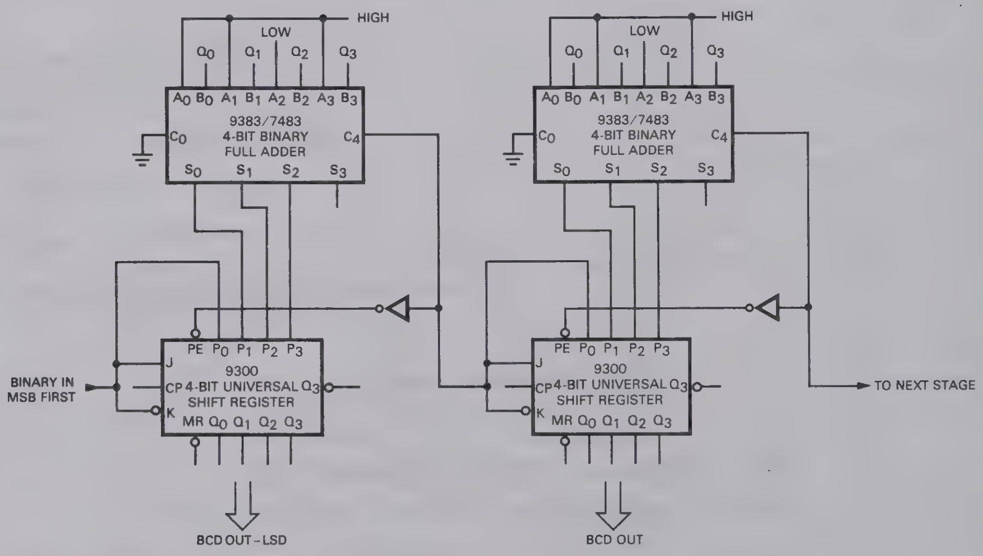
COMBINING CIRCUIT X



The parallel BCD-to-binary converter shown uses four 9383 4-bit ripple carry adders to sum all the binary equivalents of the 12 bits in a 3-digit BCD number and generates a 10-bit binary number.

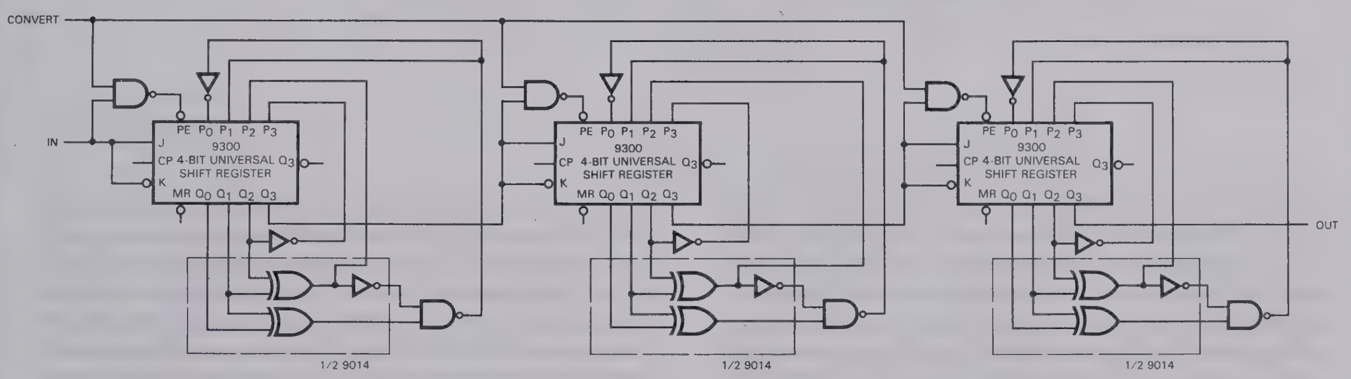
As indicated in the table, there are four inputs to the binary eight. This would normally require a considerably more complex adder structure, but since the BCD bits of weight four and eight are mutually exclusive, they can be ORed outside of the adder array and the eight can be split up into two fours. The diagram shows several different implementations for active High and active Low operands. Carry lookahead adders (9340) can be used for faster operation. This method is practical for three to four digits (four digits require 10 adders). Beyond this, the complexity of the adder structure is prohibitive.

BIT SERIAL BINARY-TO-BCD CONVERTER


HIGH
LOW
Q0
Q1
Q2
Q3
A0 B0 A1 B1 A2 B2 A3 B3
9383/7483
4-BIT BINARY
FULL ADDER
C0
C4
S0 S1 S2 S3
PE P0 P1 P2 P3
9300
4-BIT UNIVERSAL
SHIFT REGISTER
J
CP
K
Q3
MR Q0 Q1 Q2 Q3
BINARY IN
MSB FIRST
BCD OUT – LSD
BCD OUT
TO NEXT STAGE

The reverse of the BCD-to-binary algorithm is used for binary-to-BCD conversion. The binary word is shifted, most significant bit first, into a shift register consisting of several series-connected 9300s. Each shift doubles the contents of the registers in terms of BCD notation. Therefore, a correction is required whenever any of the 4-bit registers contains a number greater than four, which when shifted generates a non-BCD code. This correction is performed by adding three to the contents of the register and inserting the sum one bit downstream into the parallel data inputs. By adding 11 and then ignoring the most significant bit, the same 4-bit adder also detected whether or not the correction is necessary. A binary number is completely converted when its LSB has been shifted in, but the shift register must be long enough to hold the BCD result, always longer than the binary number. This circuit can be used for any number of bits and digits. It requires only one 9300 4-bit shift register, one 9383 4-bit adder, and one inverter for each resulting BCD digit.

SERIAL IN/SERIAL OUT BCD-TO-BINARY CONVERTER

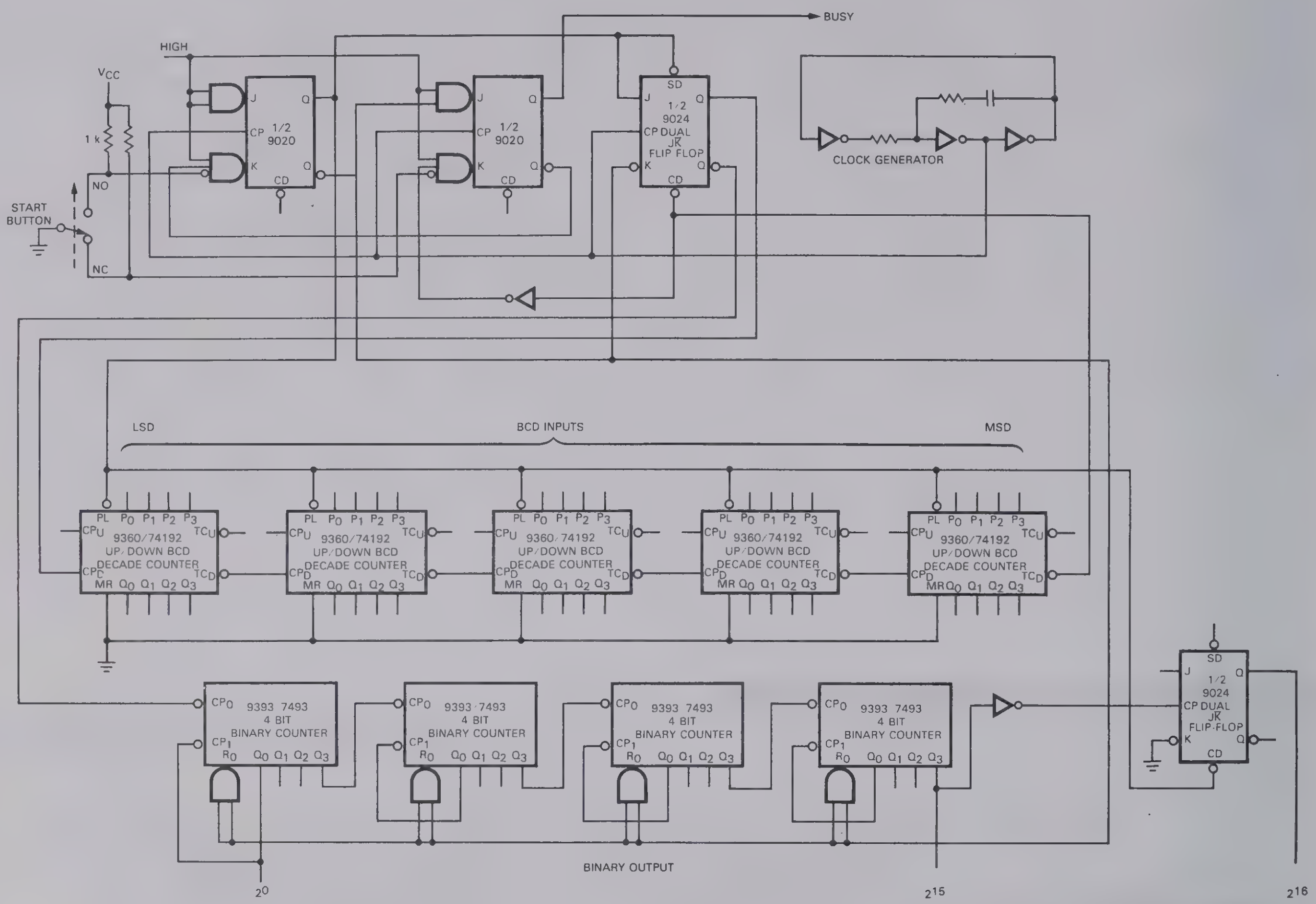


A well-known algorithm generates the binary equivalent of a BCD number by repeatedly dividing it by two. The series of least significant bits generated is the binary output, least significant first. This algorithm can be implemented with 9300 shift registers and some gates or adders.

When a BCD number is stored in the 9300 shift register with its LSB in the Q_3 stage, a right shift effectively divides it by two. A problem arises if the LSB of the more significant digit is a one, implying a value of 10 with respect to the first digit. Shifting this one into the Q_0 position changes the 10 to an eight, instead of dividing it by two. To correct for this, a three must be subtracted from the new contents of the 9300 register. The circuit shown provides a gate-minimized implementation of this algorithm using the parallel inputs of the 9300 for the correction. It converts a 4-digit (≤ 9999) BCD number into its 14-bit binary equivalent. Operation is started by bit-serially shifting in the three least significant BCD digits (LSB of the LSD first) while the Convert input is Low. The actual conversion starts when the three digits have been shifted in and the LSB of the most significant digit is being applied to the serial input. At this point, the Convert input is made High, activating the three correction networks whenever there is a one to be shifted into any of the registers. The next 14 clock pulses shift out the binary result, LSB first. This circuit can be used for any number of digits. It requires only one 4-bit shift register with a conversion network for each decimal digit except the MSD.

CODE CONVERSION

BCD-TO-BINARY CONVERTER USING COUNTERS


BUSY
HIGH
V_{CC}
1 k
NO
START BUTTON
NC
J
CP
K
Q
CD
1/2 9020
SD
1/2 9024 CP DUAL JK FLIP FLOP
CLOCK GENERATOR
LSD
BCD INPUTS
MSD
PL P_0 P_1 P_2 P_3
CP_U
TC_U
9360/74192 UP/DOWN BCD DECADE COUNTER
CP_D
TC_D
MR Q_0 Q_1 Q_2 Q_3
CP_0
CP_1
9393 7493 4 BIT BINARY COUNTER
R_0 Q_0 Q_1 Q_2 Q_3
BINARY OUTPUT
2^0
2^{15}
2^{16}

FOR BINARY TO BCD CONVERTER REPLACE 9360s WITH 9366s AND 9393/7493s WITH 9390/7490s.

Counters can be used for binary-to-BCD or for BCD-to-binary conversion, if sufficient time is available. The code to be converted is loaded into a counter, which is then counted down while another counter is counted up. When the first counter reaches zero the second counter has reached the original numeric value represented in the desired code. This method is easily expanded and can also be used for mixed modulo codes (like 6/10 for minutes and seconds, 36/10 for degrees of angle). It is also very easy to perform accumulation.

The circuit above is a BCD-to-binary converter capable of taking a 5-digit BCD number and converting it into a 17-bit binary number in less than 33 ms using a clock generator running at 6 MHz. The circuit, as shown, is completely self-contained including a clock generator and a debounce/edge detect circuit for use with a push button to initiate the conversion. Only 11 integrated circuit packages are required.

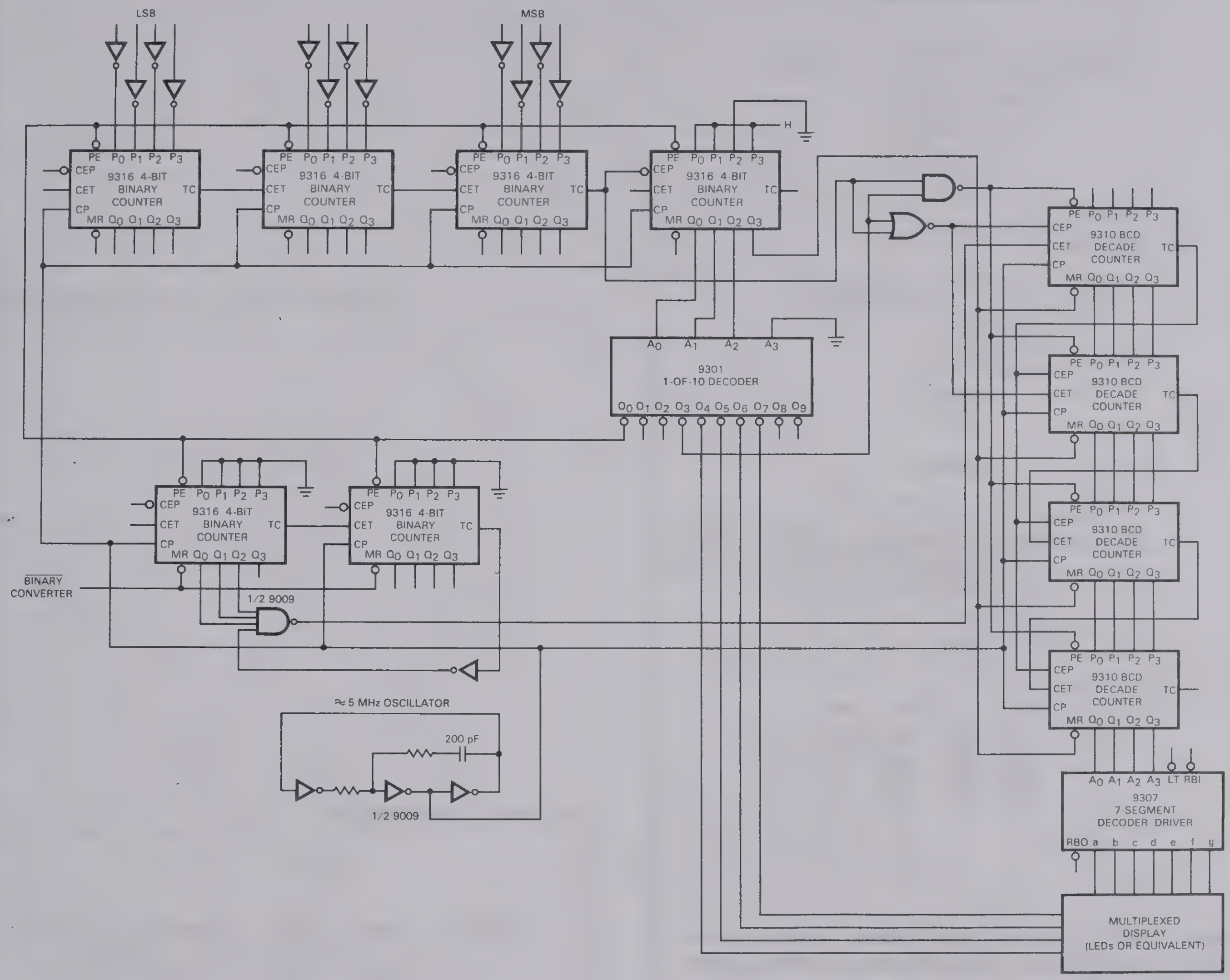
A 9020 dual flip-flop is used to debounce and detect a new closure on the start button and to generate a pulse one clock period wide which loads the BCD number into a decimal down counter and clears a binary up counter. The two 9020 flip-flops are normally in an idle state with the first flip-flop set (Q = High) and the second reset (Q = Low). Depressing the start button causes the first flip-flop to set, activating the asynchronous Load and Reset inputs on the respective counters and asynchronously setting a 9024 flip-flop used to drive the clock inputs on both counters. While a clear direct signal initially exists to this flip-flop, it quickly disappears as the clock input to the down counter is forced High and the counter is loaded with a new number.

The next clock pulse causes both 9020 flip-flops to set, removing the counter load and reset signals, generating a "busy" signal, and enabling the 9024 flip-flop to toggle on subsequent clock pulses. The two counters count, one down and the other up, at half the clock generator frequency until the decimal down counter reaches zero.

After the down counter has reached zero and the 9024 flip-flop has again reset, the $\overline{TC}_D$ output from the counter goes Low holding the flip-flop in the reset state and thus locking the counters. $\overline{TC}_D$ = Low also enables the second 9020 to reset if the start button has been released. The push button must be released and depressed again to initiate a new conversion.

The converter provides the correct binary output for any number within the range of the counter including zero.

BINARY ANGLE TO BCD CONVERTER WITH DISPLAY


LSB
MSB
H
9316 4-BIT BINARY COUNTER
PE P0 P1 P2 P3
CEP
CET
CP
TC
MR Q0 Q1 Q2 Q3
9310 BCD DECADE COUNTER
9301 1-OF-10 DECODER
A0 A1 A2 A3
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
BINARY CONVERTER
1/2 9009
≈ 5 MHz OSCILLATOR
200 pF
9307 7-SEGMENT DECODER DRIVER
A0 A1 A2 A3 LT RBI
RBO a b c d e f g
MULTIPLEXED DISPLAY (LEDs OR EQUIVALENT)

This converter can perform a code conversion and display the result continuously in binary coded decimal (BCD). The converter, as shown, operates in either of two modes, binary-to-BCD or binary angle-to-BCD. In the binary angle code, the most significant bit represents 180° (half circle), the next 90° (quarter circle), the third 45° (eighth circle), etc. The converter accepts 12 bits of this code and converts it to degrees and tenths of degrees (0.0° to 359.9°). The converter is a self-contained circuit requiring only 17 packages capable of running at a 5 MHz clock rate. The BCD output is displayed in an economical manner by multiplexing the four digits.

The circuit operates by automatically loading the complement of the input code into the binary counter at the beginning of the cycle while the decimal counter is cleared. The binary counter reaches terminal count after n clock pulses where n = binary input number. In the binary conversion mode, the decade counter also counts n times and thus reaches the BCD equivalent of the binary input. If the ability to convert angles to BCD is not needed, the two lower 9316s and two gates are not needed.

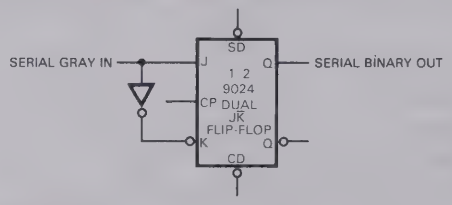
After the conversion, the BCD data is displayed by multiplexing the digits, most significant digit first. The four stage binary counter slowly shifts the digits through the decimal counter while enabling one display digit at a time by counting 4096 clock pulses per digit. After displaying the least significant digit, the cycle is repeated.

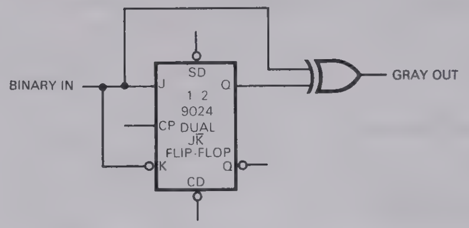
In the binary angle-to-BCD mode, the decimal counter is incremented at a slower rate than the binary counter to adjust for the weights of the binary angle bits entered in the binary counter. The most significant bit in a binary-to-BCD conversion has a weight of 2'' or 2048, but in this binary angle-to-BCD conversion it has a weight of 180° or 1800 tenths of a degree. The BCD counter is therefore incremented at 1800/2048 or 225/256 the rate of the binary counter by inhibiting the decimal counter 31 times during every 256 clock pulses. This rate multiplier function is performed by two 9316s (modulo 256 counter) and two gates decoding every eighth state except TC of the counter for a total of 31 states. These evenly distributed inhibit pulses minimize the conversion error.

CODE CONVERSION

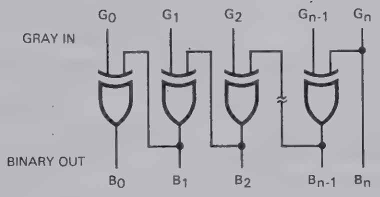
GRAY CODE CONVERSIONS

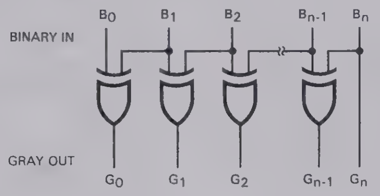
SERIAL, MSB FIRST





PARALLEL





Binary codes are not particularly suited for electrical or electro-optical encoder systems (angular position shaft encoders, etc.) because a movement from one state to the next often results in more than one bit change, i.e., from seven to eight, the binary code changes from 0111 to 1000. Such bit changes can never really be simultaneous, so the encoder always generates erroneous transient codes when switching between certain positions. This problem is avoided with a Gray code because only one bit changes between adjacent states. The Gray code is a non weighted code and awkward for other applications. It must be converted to binary or BCD before any arithmetic can be performed.

In Gray-to-binary *serial* conversion, a flip-flop that toggles for every one performs the conversion. The most significant bit, however, must come in first. Gray to-binary *parallel* conversion is performed by a series of Exclusive OR gates.

In binary-to-Gray *serial* conversion, a flip-flop acts as a 1-bit delay element and an Exclusive OR gate is used between the present and the previous binary bit. Note that, in this case as well as in Gray-to-binary serial conversion, the most significant bit must come in first. Binary-to-Gray *parallel* conversion is performed by a series of Exclusive OR gates.

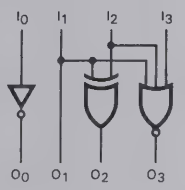
EXCESS 3 GRAY CODE

DECIMAL	BINARY	GRAY	X3 BINARY	X3 GRAY
0	0000	0000	0011	0010
1	0001	0001	0100	0110
2	0010	0011	0101	0111
3	0011	0010	0110	0101
4	0100	0110	0111	0100
5	0101	0111	1000	1100
6	0110	0101	1001	1101
7	0111	0100	1010	1111
8	1000	1100	1011	1110
9	1001	1101	1100	1010
10	1010	1111		
11	1011	1110		
12	1100	1010		
13	1101	1011		
14	1110	1001		
15	1111	1000		

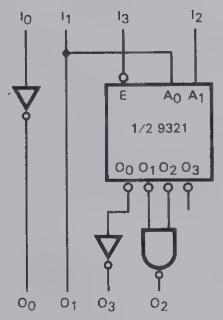
Decimal systems use Excess 3 Gray Code because this code has the feature of changing only one bit at a time even on a nine-to-zero transition. Excess 3 Gray Code is detected or generated in the same manner as Gray codes, but adding a three to the binary value for binary-to-Excess 3 conversion and subtracting a three (i.e., adding binary 13) from the binary value for Excess 3-to-binary conversion.

CODE CONVERSION

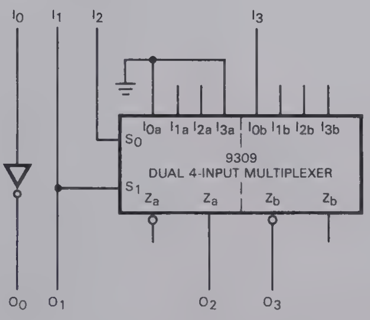
GENERATING NINES COMPLEMENTS



I0
I1
I2
I3
O0
O1
O2
O3



I0
I1
I3
I2
E
A0
A1
1/2 9321
O0 O1 O2 O3
O0
O1
O3
O2

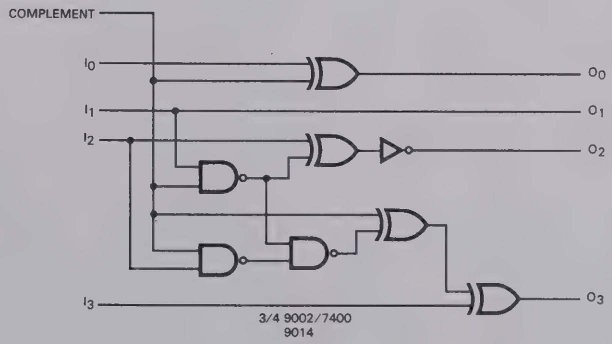


I0
I1
I2
I3
S0
I0a I1a I2a I3a
I0b I1b I2b I3b
9309
DUAL 4-INPUT MULTIPLEXER
S1
Za
Za
Zb
Zb
O0
O1
O2
O3

The one complement of a binary number is easily generated by inverting each bit. The equivalent in a decimal (BCD) system, nine complement, is not that easy. These three circuits convert a 1-digit BCD input into its nine complement. They use about one equivalent gate or MSI package per digit (decade).

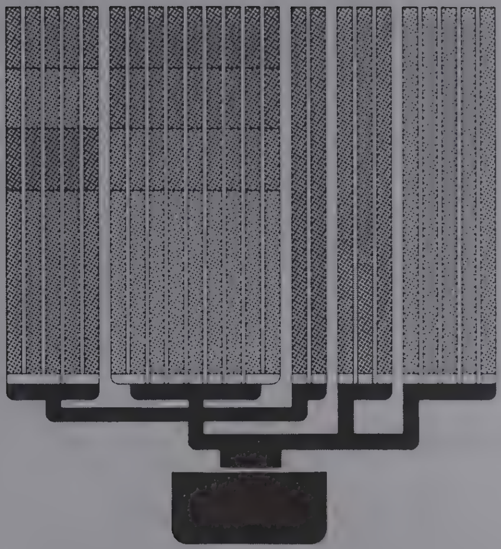
CONTROLLED NINE COMPLEMENT CIRCUIT USING TWO GATE PACKAGES

I_0	I_1	I_2	I_3	Compl.	O_0	O_1	O_2	O_3
X	X	X	X	L	$=I_0$	$=I_1$	$=I_2$	$=I_3$
L	L	L	L	H	L	L	L	H
H	L	L	L	H	L	L	L	H
L	H	L	L	H	H	H	H	L
H	H	L	L	H	L	H	H	L
L	L	H	L	H	H	L	H	L
H	L	H	L	H	L	L	H	L
L	H	H	L	H	H	H	L	L
H	H	H	L	H	L	H	L	L
L	L	L	H	H	H	L	L	L
H	L	L	H	H	L	L	L	L



COMPLEMENT
I0
I1
I2
I3
O0
O1
O2
O3
3/4 9002/7400
9014

This controlled nine complement circuit, using two gate packages, either generates the nine complement or transfers the BCD inputs through unchanged.



Latches

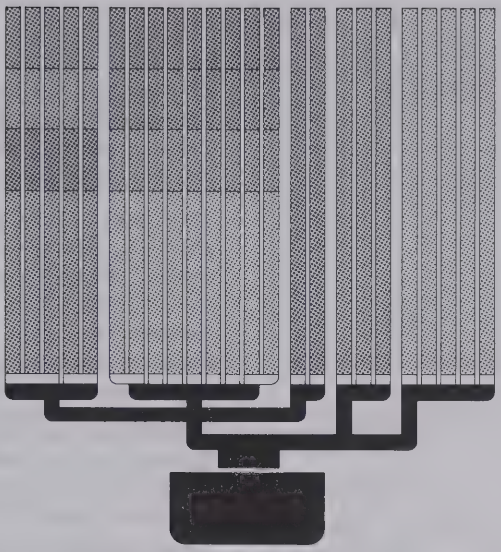
- Register Selection
- Register Using Latches as M/S Flip-Flops
- Holding Register in Counting and Display
- Multiplexing to a Data Bus
- Contact Bounce Eliminator
- One or Zero Catching
- Count and Hold Display System
- 16-Word by 4-Bit Memory
- 64-Word by 4-Bit Memory
- Demultiplexing and Decoding
- Output Interface Buffer
- Successive Approximation A/D Converter
- Serial-to-Parallel Converter
- 4096-Bit, 15 MHz Serial Memory
- Noise and Contact Bounce Suppression

LATCH SELECTION GUIDE

*$\overline{R}$ input used as $\overline{D}$ input when $\overline{S}$ is grounded

**With $\overline{SLE}$ input grounded

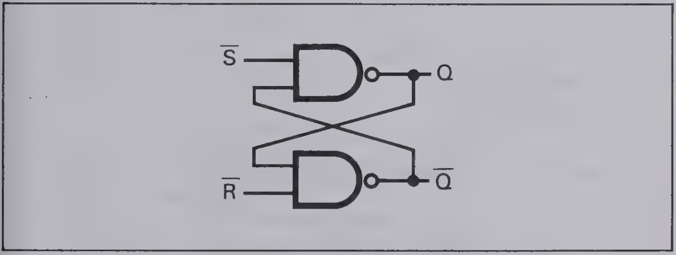
Function	Device	Data Inputs	Common Clear	Enable Inputs (Level)	Required Enable Pulse Width ns (typ)	Enable to Q Delay, ns (typ)	Data to Q Delay, ns (typ)	Power Dissipation mW (typ)
4-Bit R-S Latch	9314	4 x ($\overline{R}_1\overline{S}_1$)	L	1 (L)	12	18	18	175
	93L14	4 x ($\overline{R}_1\overline{S}_1$)	L	1 (L)	30	51	45	50
4-Bit D Latch	9314	4 x D*	L	1 (L)	12	18	18	175
	93L14	4 x D*	L	1 (L)	30	51	45	50
Dual 2-Bit D Latch	9375, 7475	4 x D		2 (H)	20	16	16	160
	9377, 7477	4 x D		2 (H)	20	16	16	160
Dual 4-Bit D Latch	9308 (74116)	8 x D	2 x L	2 x 2 AND (L)	15	25	20	310
	93L08	8 x D	2 x L	2 x 2 AND (L)	30	32	32	100
8-Bit Addressable Latch	9334	1 x D	L	1 (L) 3 Address Bits	11	18	28	280
8-Bit Multiport Register**	9338	1 x D		1 (L) 9 Address Bits	7	35	35	425



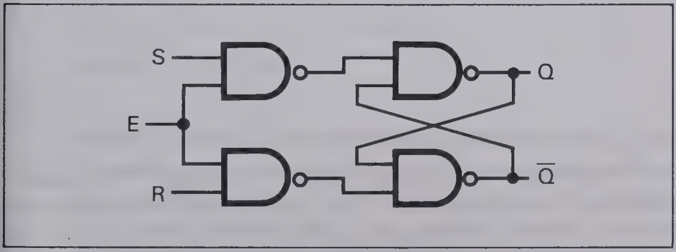
LATCHES

INTRODUCTION

Latches are the simplest data storage devices. The basic latch circuit consists of two cross-coupled gates, usually NAND gates.

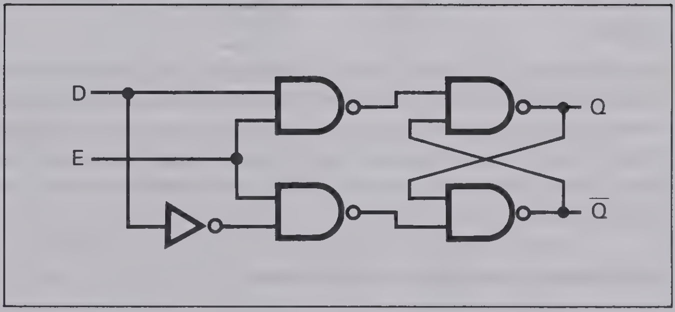


A Low on the $\overline{S}$ input sets the latch (Q High, $\overline{Q}$ Low), a Low on the $\overline{R}$ input resets it. When both inputs are High the latch stays in its previous state.



Using two more gates, the latch can be strobed or enabled. When the Enable input is High, the S or R input affects the latch, when E is Low the latch is not affected by the inputs.

By generating R = $\overline{S}$ (using an additional inverter) the latch is changed to a D type. The Q output follows the D input as long as E is High, but the latch stays locked when E goes Low.



Latches are transparent, i.e., when enabled the output can change as the inputs change. Latches should not, therefore, be used in applications where the latch output feeds back into the input, since this could create a race condition (oscillation). Registers should be used in such cases.

Memories are latch arrays with addressing structures common for input and output and are covered in the memory section.

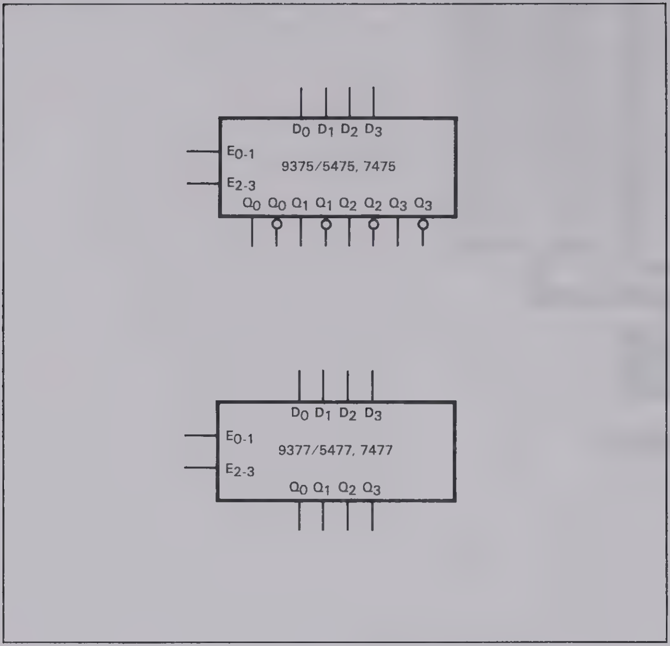
Five different latch circuits are presently available in the Fairchild family of TTL MSI circuits. The selection guide summarizes their capabilities and features.

Latches can be grouped in three catagories:

- 4-bit and dual 4-bit D-latches with parallel access to all inputs and outputs. (9308, 9314, 9375, 9377)
- 4-bit RS latches (9314)
- 8-bit addressable latch arrays (9334)

LATCHES

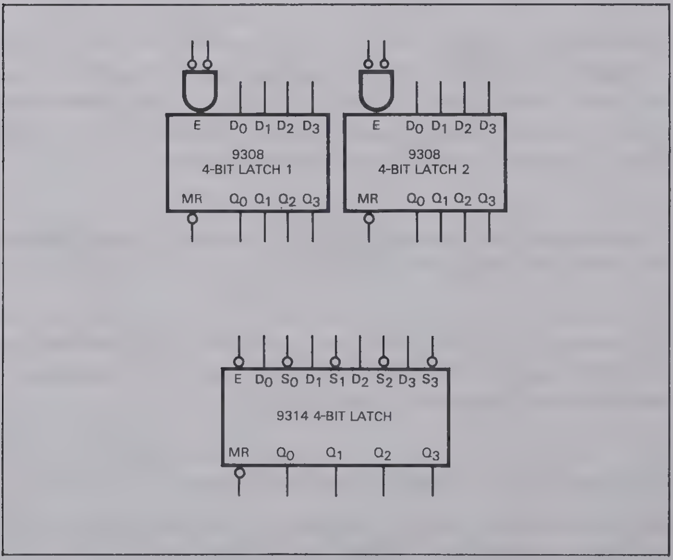
9375/5475, 7475 And 9377/5477, 7477 4-Bit Latches


D_0 D_1 D_2 D_3
$E_{0\text{-}1}$
9375/5475, 7475
$E_{2\text{-}3}$
Q_0 $\overline{Q}_0$ Q_1 $\overline{Q}_1$ Q_2 $\overline{Q}_2$ Q_3 $\overline{Q}_3$
D_0 D_1 D_2 D_3
$E_{0\text{-}1}$
9377/5477, 7477
$E_{2\text{-}3}$
Q_0 Q_1 Q_2 Q_3

The 9375 and 9377 are 4-bit D-type storage latches separated into two 2-bit sections, each with an active High Enable. Data enters the latches when the Enable is High and is stored when the Enable is Low. The 9375 has dual polarity outputs; the 9377 14-lead version does not have the complement outputs.

Both devices are useful for data storage in groups of two bits rather than the usual four. The 9375 is useful when complementary outputs are required. However, their high Enable drive requirements, active High Enable polarity, lack of reset (clear), and non-standard power lead locations must be considered.

9308 And 9314 Dual 4-Bit Latches


E D_0 D_1 D_2 D_3
9308
4-BIT LATCH 1
MR Q_0 Q_1 Q_2 Q_3
E D_0 D_1 D_2 D_3
9308
4-BIT LATCH 2
MR Q_0 Q_1 Q_2 Q_3
E D_0 S_0 D_1 S_1 D_2 S_2 D_3 S_3
9314 4-BIT LATCH
MR Q_0 Q_1 Q_2 Q_3

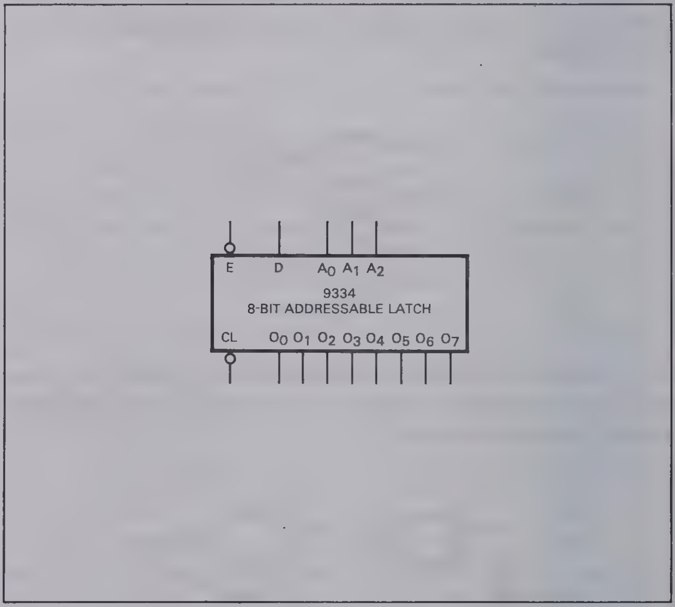
The 9308 and 9314 are dual 4-bit parallel D type storage latches which both offer asynchronous overriding Reset (clear), active Low Enable compatible with other MSI decoder elements, on-chip buffering to reduce input loading and standard power lead locations. Both the 9308 and 9314 include circuitry to eliminate glitches or logic transients when the Enable is removed.

The 9308 is a 24-lead device with two separate 4-bit latch sections, each with an overriding Master Reset and a 2-input active Low AND Enable. The AND Enable increases flexibility for controlling data entry. One input can be tied to the clock disabling the latch during logic transitions, the other can be used for latch selection, or both Enables can be used for expansion.

The 9314 is a 4-bit latch with active Low Enable and Reset. Each of the four latches can operate as gated active Low Set/ Reset latches, (with reset override) or as D-type storage latches ($\overline{S}$ input to ground). Thus, the 9314 can reduce inventories because it is equally appropriate in both RS and D latch applications. In D applications it may be preferred over the 9308 due to the smaller space required by two 16-lead packages (compared to the 24-lead 9308 package). The 9314 is the only 4-bit gated RS latch currently available. It offers an Enable for controlling the RS latch and has no undefined states or unallowed input conditions (typical with RS latches built with cross-coupled gates).

In general, the 9308 and 9314 are the best suited devices in the 4-bit and dual 4-bit latch category for parallel data storage. They are compatible with other MSI, incorporate more features and in most cases, require less additional logic.

9334 8-Bit Addressable Latch


E D A_0 A_1 A_2
9334
8-BIT ADDRESSABLE LATCH
CL O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7

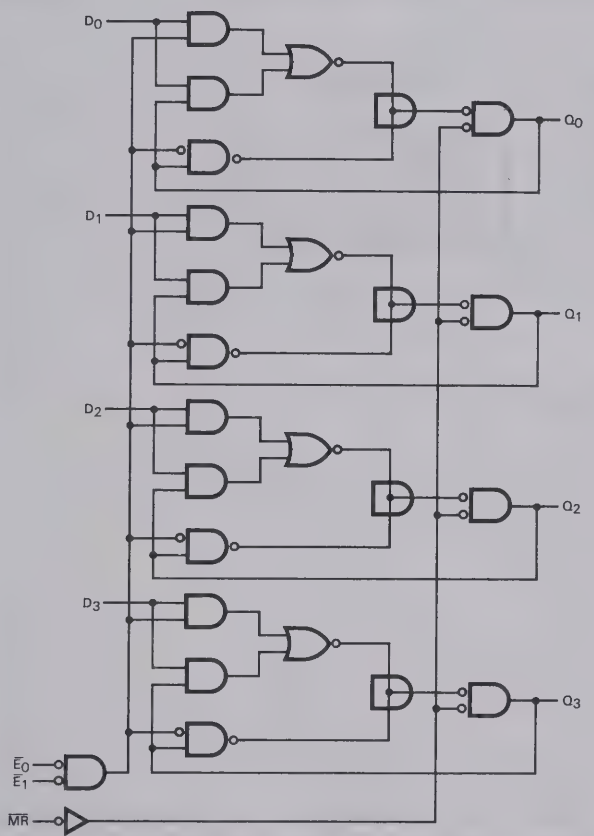
The 9334 latch offers high density storage. The outputs from eight latches are available and a common active Low Enable and Clear are provided in a 16-lead package. In addition to storing information, the 9334 also can serve as a 1-of-8 decoder or demultiplexer. It has a single D-type data input; three Address inputs select and a fourth input enables writing into only one of the eight latches. It is valuable for random or sequential bit by bit data storage applications where all stored bits must be continuously available. The 9334 also is efficient for storing parallel data in arrays of eight or more words.

9308 DUAL 4-BIT LATCH

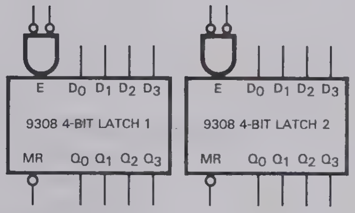
DESCRIPTION AND OPERATION

LEADS		LOADING
D_0, D_1, D_2, D_3	Parallel Latch Inputs	1.5 UL
$\bar{E}_0, \bar{E}_1$	AND Enable Inputs	1 UL
$\overline{MR}$	Master Reset Input	1 UL
Q_0, Q_1, Q_2, Q_3	Parallel Latch Outputs	9 UL

CHARACTERISTICS	
TYPICAL DELAYS	$\bar{E}$ to Output 20 ns D to Output 15 ns
PACKAGE	24-Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	340 mW



NOTE: ONLY ONE 4-BIT LATCH SHOWN.



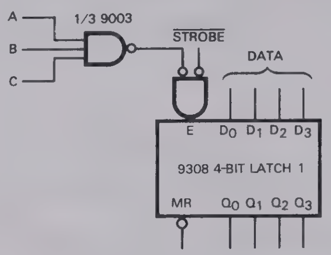
$\overline{MR}$	$\bar{E}_0$	$\bar{E}_1$	D	Q_N	OPERATION
H	L	L	L	L	Data Entry
H	L	L	H	H	Data Entry
H	L	H	X	Q_{N-1}	Hold
H	H	L	X	Q_{N-1}	Hold
H	H	H	X	Q_{N-1}	Hold
L	X	X	X	L	Reset

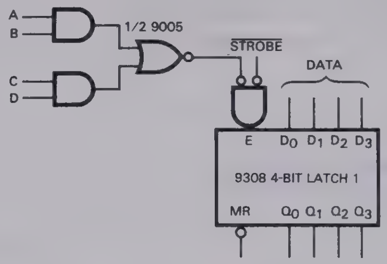
The 9308 dual 4-bit latch is monolithic MSI circuit designed for general purpose storage applications in high speed digital systems. It is a high speed, versatile and economical storage building block particularly suitable for use with the 9301 1-of-10 decoder, the 9311 1-of-16 decoder and the 9309 and 9312 multiplexers to form high speed storage systems.

The logic diagram for one block of four latches is shown. Data is entered in parallel into the four latches when the Enable inputs are both at the Low logic level. The information appears at the outputs within 20 ns of receipt of the Enable pulse and is held in each latch by the feedback logic. Each latch incorporates an additional AND gate to prevent glitches or logic spikes when the Enable inputs are removed. A common Reset which overrides all other input conditions is provided for the four latches, holding their outputs Low while it is active. The Master Reset and Enable inputs are buffered, presenting one TTL input load. Typically, one active Low input Enable is used to select the latch block and the other is used for the clock or strobe input. Latch operation is summarized by the truth table.

9308 DUAL 4-BIT LATCH APPLICATIONS

REGISTER SELECTION

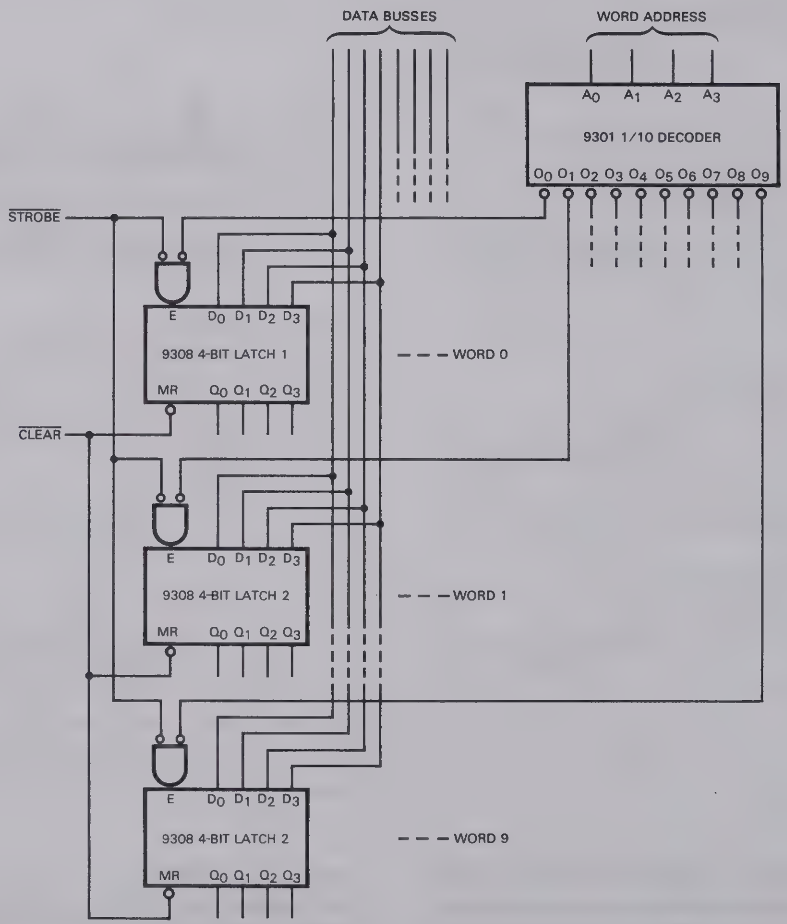

A
B
C
1/3 9003
STROBE
DATA
E D0 D1 D2 D3
9308 4-BIT LATCH 1
MR Q0 Q1 Q2 Q3


A
B
C
D
1/2 9005
STROBE
DATA
E D0 D1 D2 D3
9308 4-BIT LATCH 1
MR Q0 Q1 Q2 Q3

On the left is a block of four latches used with a 3-input NAND gate feeding into one input on the active Low Enable and a strobe signal on the remaining active Low Enable. This logic allows data entry into the latch block for an AND condition A • B • C • $\overline{\text{Strobe}}$

on the right is the logic for several AND conditions (A • B or C • D) using an AND/OR/INVERT gate. These examples indicate the logic power of the active Low inputs on the Enable gate.

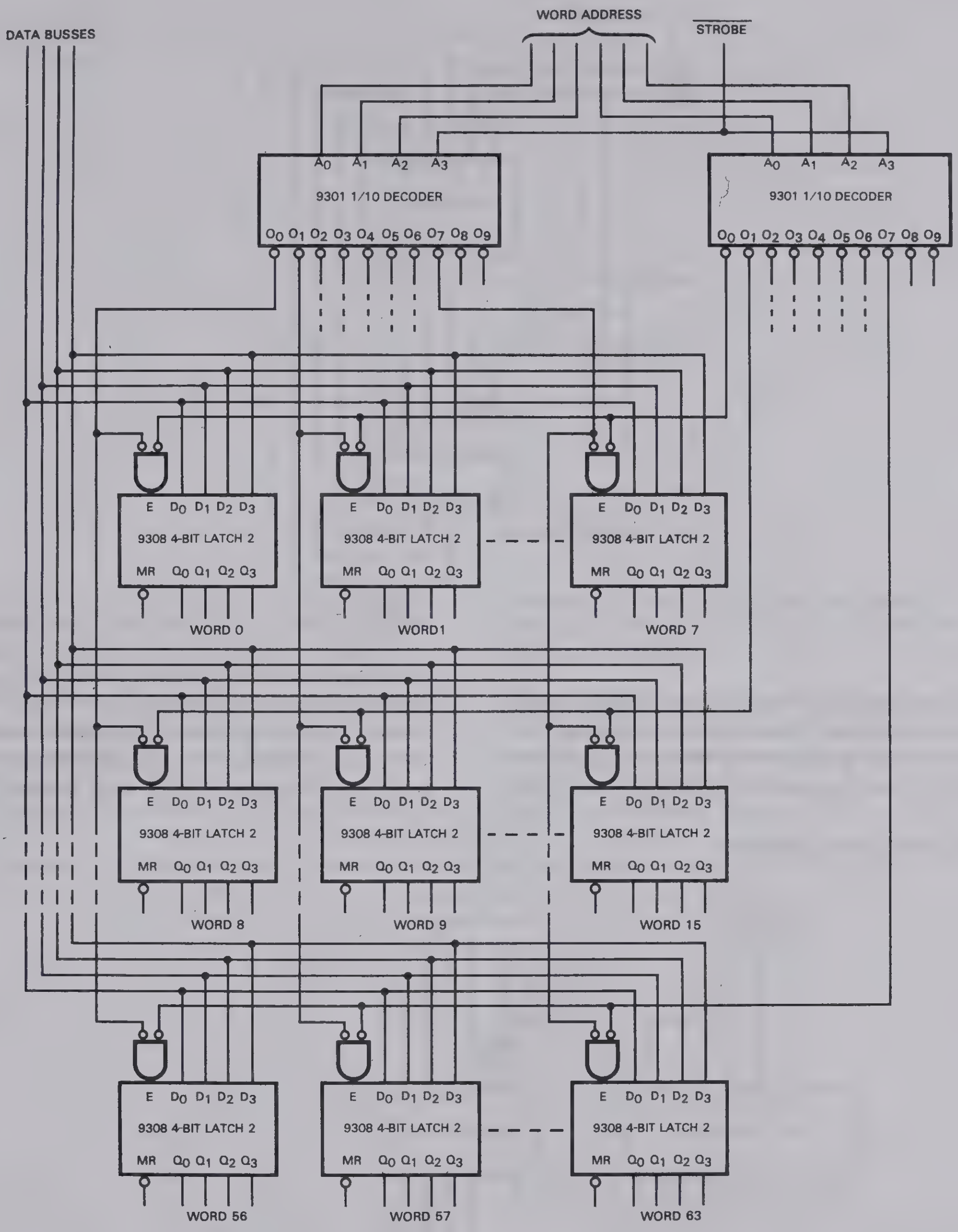
REGISTER SELECTION


DATA BUSSES
WORD ADDRESS
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
STROBE
CLEAR
E D0 D1 D2 D3
9308 4-BIT LATCH 1
MR Q0 Q1 Q2 Q3
WORD 0
E D0 D1 D2 D3
9308 4-BIT LATCH 2
MR Q0 Q1 Q2 Q3
WORD 1
E D0 D1 D2 D3
9308 4-BIT LATCH 2
MR Q0 Q1 Q2 Q3
WORD 9

The 9308 was specifically designed to work with the other members of the 9300 MSI family, and the active Low Enable AND inputs prove useful for logic designs in conjunction with the 9321 1-of-4, the 9301 1-of-10 and the 9311 1-of-16 decoders. This diagram shows a register selection scheme in which information is presented on parallel busses. The register receiving data is selected by a 9301 decoder. The only condition necessary for correct entry is that register address inputs to the decoder must be stable before the registers are strobed. Since only a single unit load is presented by a 9308 Enable input, one 9301 decoder can control up to 10 registers containing up to 40 bits each.

9308 DUAL 4-BIT LATCH APPLICATIONS

MATRIX REGISTER



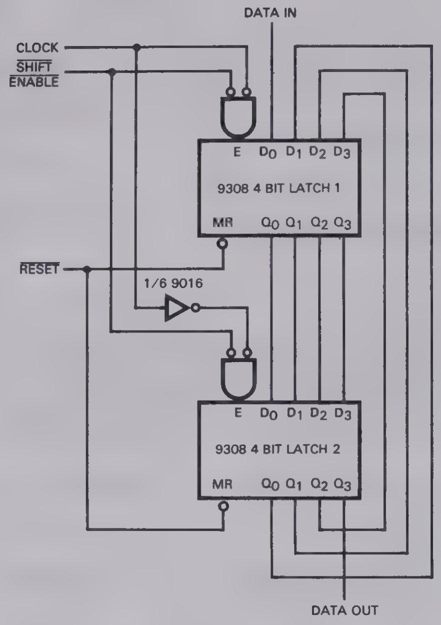
DATA BUSSES
WORD ADDRESS
STROBE
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
E D0 D1 D2 D3
9308 4-BIT LATCH 2
MR Q0 Q1 Q2 Q3
WORD 0
WORD1
WORD 7
WORD 8
WORD 9
WORD 15
WORD 56
WORD 57
WORD 63

A circuit for display or special storage applications is shown. Two decoders select the register to receive information from a set of common bus lines. The matrix above has 64 registers in an 8 x 8 matrix. Outputs 0-7 of each decoder are used in a coincident select scheme with one decoder's outputs common in the X direction and the other decoder's outputs common in the Y direction. Outputs O_8 and O_9 of the 9301 decoders are not used; therefore the most significant A_3 input on the decoders can be employed as a strobe input. The Low level fan out of the decoder is 10 unit loads. As shown, each decoder output drives eight enable inputs. For systems with larger word length, buffers should be provided, (requiring two extra logic levels) or the more economical method of duplicating the decoding scheme should be used. The size of the matrix can be extended to 10 x 10 words by using the A_3 inputs of the decoders, but to accept a strobe input, 2-input gates must be added to both the A_2 and A_3 inputs of one of the decoders. These gates force the code for 12 or a greater number on the inputs of the decoder when a strobe is not present, forcing all decoder outputs High.

The 9311 1-of-16 decoder can also be used in a matrix format. However, since the fan out at the Low logic level is 10, a 256-word matrix must share the load between two 9311 decoders. Various other decoding methods to select a single register from a group of registers can also be used with the 9301 and 9311 decoders.

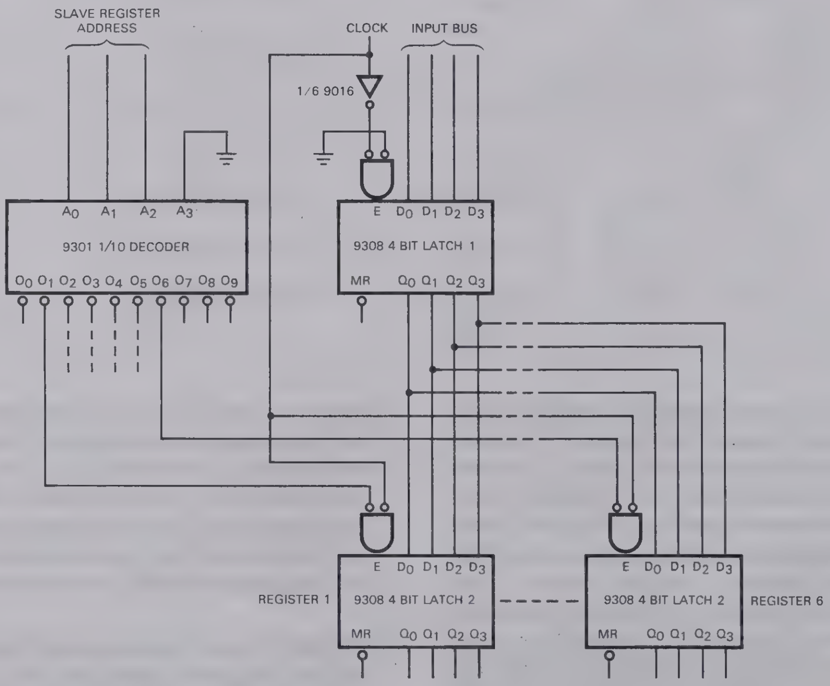
9308 DUAL 4-BIT LATCH APPLICATIONS

REGISTER USING LATCHES AS M/S FLIP-FLOPS


DATA IN
CLOCK
SHIFT ENABLE
E D0 D1 D2 D3
9308 4 BIT LATCH 1
MR Q0 Q1 Q2 Q3
RESET
1/6 9016
E D0 D1 D2 D3
9308 4 BIT LATCH 2
MR Q0 Q1 Q2 Q3
DATA OUT

The 9308 can be used as a bank of four master/slave flip-flops by providing two Clock or Enable signals, one for the latch block acting as a master and another for the slave. A single inverting gate can generate these two clock phases from a single clock input. This diagram shows the dual latch as a 4-bit shift register using this two phase clock system. This scheme can be extended providing one master latch and several slave latches.

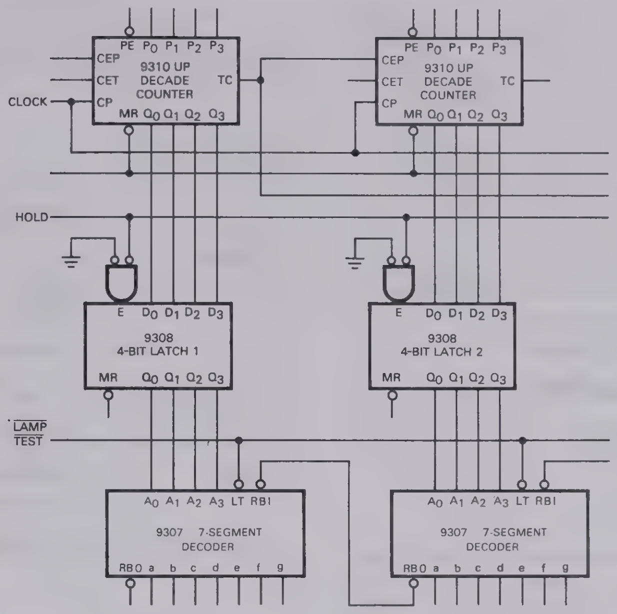
SINGLE MASTER/MULTIPLE SLAVE


SLAVE REGISTER ADDRESS
CLOCK
INPUT BUS
1/6 9016
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
E D0 D1 D2 D3
9308 4 BIT LATCH 1
MR Q0 Q1 Q2 Q3
REGISTER 1
E D0 D1 D2 D3
9308 4 BIT LATCH 2
MR Q0 Q1 Q2 Q3
E D0 D1 D2 D3
9308 4 BIT LATCH 2
MR Q0 Q1 Q2 Q3
REGISTER 6

One 4-bit latch acting as the master drives six 4-bit latches acting as slaves. A decoder selects the slave to receive information from the master. This method is very economical in terms of hardware and offers considerable savings over a multiple flip-flop register approach. The receiving slave is determined by the 9301 decoder acting as a demultiplexer for the strobe or clock signal.

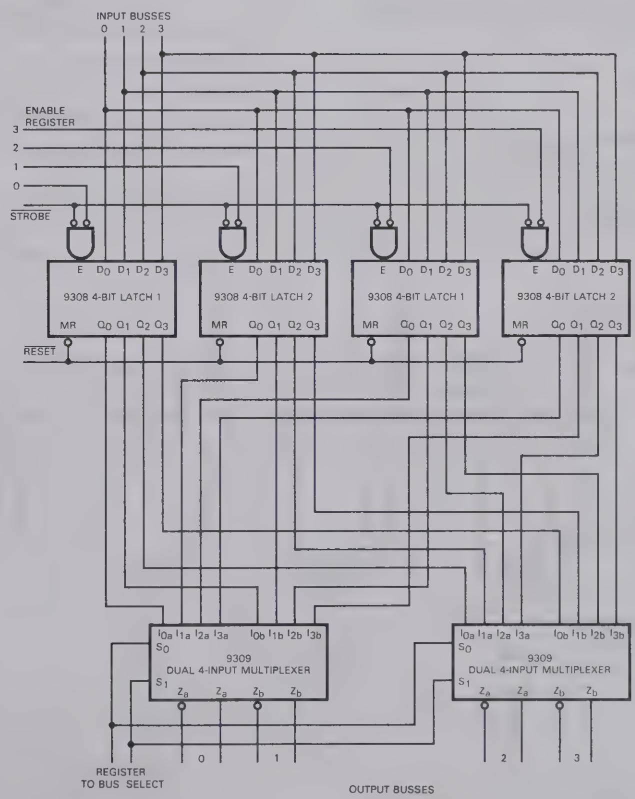
9308 DUAL 4-BIT LATCH APPLICATIONS

HOLDING REGISTER IN COUNTING AND DISPLAY


PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
HOLD
E D0 D1 D2 D3
9308 4-BIT LATCH 1
9308 4-BIT LATCH 2
MR Q0 Q1 Q2 Q3
LAMP TEST
A0 A1 A2 A3 LT RBI
9307 7-SEGMENT DECODER
RBO a b c d e f g

The dual 4-bit latch can be used as a holding register in counting operations. Two 9310 synchronous decade counters are illustrated in a typical counter design. At a specified time, the information from the counters is latched into the holding register where it is held and displayed via the 9307 7-segment decoders until the next sample time.

MULTIPLEXING TO A DATA BUS


INPUT BUSSES
0 1 2 3
ENABLE REGISTER
3
2
1
0
STROBE
E D0 D1 D2 D3
9308 4-BIT LATCH 1
9308 4-BIT LATCH 2
MR Q0 Q1 Q2 Q3
RESET
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309 DUAL 4-INPUT MULTIPLEXER
Za Za Zb Zb
REGISTER TO BUS SELECT
0 1 2 3
OUTPUT BUSSES

In addition to selecting a particular register consisting of latches, it is often necessary to multiplex the information from several latch registers onto a bus system. The 9309 and 9312 digital multiplexers are useful building blocks for this purpose. Here, four 4-bit registers are switched to the appropriate busses under control of the S_0 and S_1 inputs of the multiplexer.

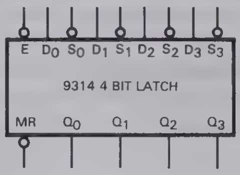
9314 4-BIT LATCH

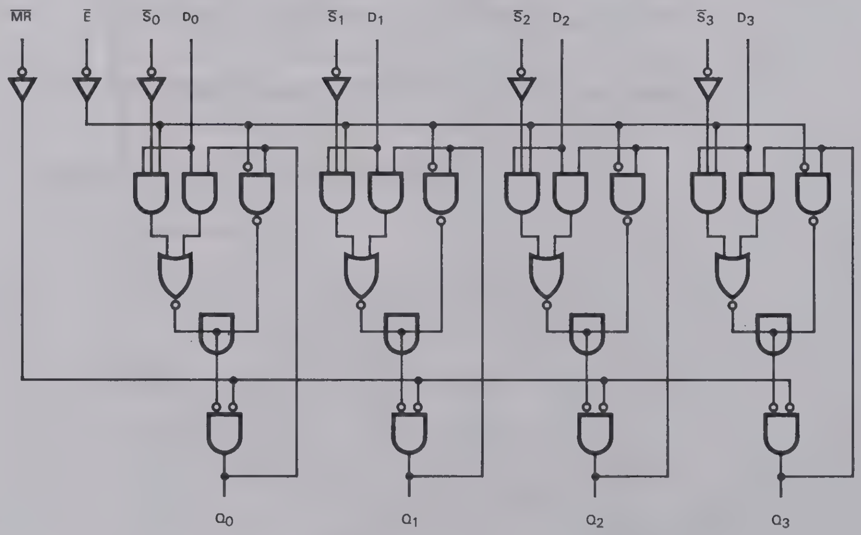
DESCRIPTION AND OPERATION

CHARACTERISTICS	
TYPICAL DELAYS	$\overline{E}$ to Q 20 ns D to Q 15 ns
PACKAGE	16-Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	175 mW

LEADS		LOADING
$\overline{E}$	Enable Input (Active Low)	1 UL
D_0, D_1, D_2, D_3	Parallel Data Inputs	1.5 UL
$\overline{S}_0, \overline{S}_1, \overline{S}_2, \overline{S}_3$	Set (Active Low) Inputs	1 UL
$\overline{MR}$	Master Reset (Active Low) Input	1 UL
Q_0, Q_1, Q_2, Q_3	Parallel Outputs	9 UL

$\overline{MR}$	$\overline{E}$	D	$\overline{S}$	Q_N	OPERATION
H	L	L	L	L	D MODE
H	L	H	L	H	
H	H	X	X	Q_{N-1}	
H	L	L	L	L	R/S MODE
H	L	H	L	H	
H	L	L	H	L	
H	L	H	H	Q_{N-1}	
H	H	X	X	Q_{N-1}	
L	X	X	X	L	RESET

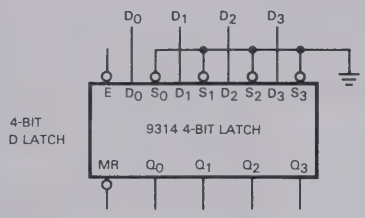


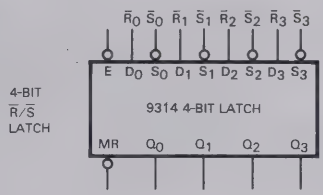


Fairchild's 9314 is a monolithic MSI circuit designed for general purpose storage applications in high speed digital systems. The 9314 is a 4-bit latch with a common active Low Enable and Master Reset. Each of the four latches can operate as a gated active Low set/reset latch with reset override, or as a D-type storage latch. The four latches have a common active Low Enable and active Low Master Reset.

9314 4-BIT LATCH APPLICATIONS

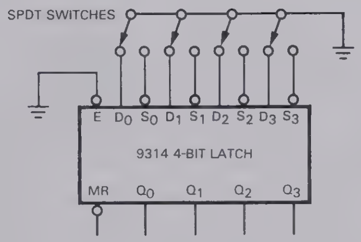
OPERATIONAL MODES


D0 D1 D2 D3
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3
4-BIT
D LATCH


$\overline{R}_0$ $\overline{S}_0$ $\overline{R}_1$ $\overline{S}_1$ $\overline{R}_2$ $\overline{S}_2$ $\overline{R}_3$ $\overline{S}_3$
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3
4-BIT
$\overline{R}/\overline{S}$
LATCH

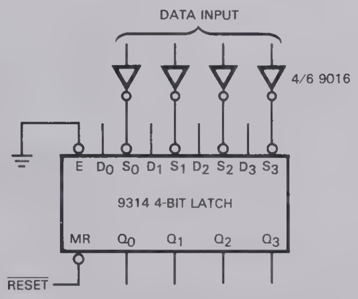
Each latch can be operated in one of two modes, (see Truth Table) and the operation and connection of the 9314 in both modes is shown above. In the active Low $\overline{RS}$ mode, the latch is reset by a Low on the D ($\overline{R}$) input and is set by a Low on the $\overline{S}$ input, effective only while the common Enable is Low. If both $\overline{S}$ and D ($\overline{R}$) inputs are Low, the D ($\overline{R}$) input dominates and the latch is reset. This avoids the undefined states normally found in RS latches. Mode D operation is obtained when the S input of a latch is held Low. While the common Enable is active, (Low) the latch output follows the D input. Information present at the latch input is stored in the latch when Enable goes High.

CONTACT BOUNCE ELIMINATOR

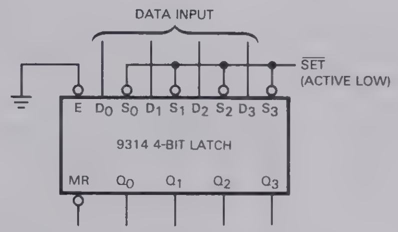

SPDT SWITCHES
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3

The 9314 can be used as a contact bounce eliminator. When a mechanical switch is closed, the contacts usually bounce a number of times before settling. This could be misinterpreted by a high speed digital circuit. The latch eliminates this problem. When the switch is in the left-hand position, the latch is reset repeatedly as bouncing occurs; when it is in the right-hand position, it is set repeatedly. The Enable, shown tied to ground, can be used to strobe the switches to store the switch pattern at any particular time.

ONE OR ZERO CATCHING


DATA INPUT
4/6 9016
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3
$\overline{\text{RESET}}$

ONE CATCHING

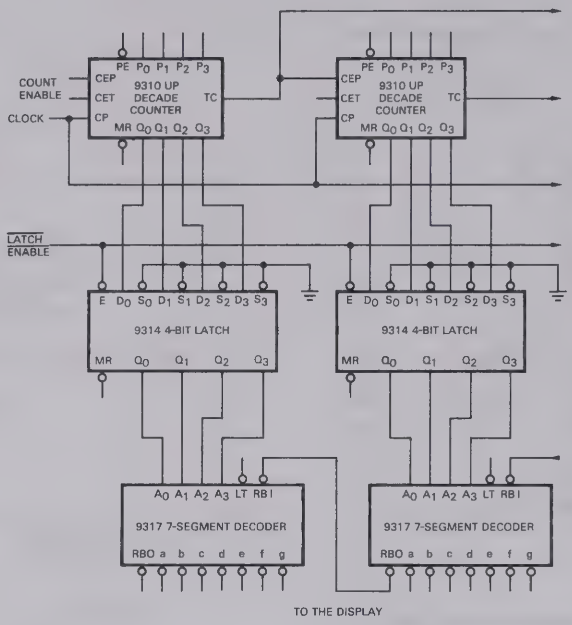

DATA INPUT
$\overline{\text{SET}}$
(ACTIVE LOW)
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3

ZERO CATCHING

This is the 9314 used as a one or zero catcher. In this application, data inputs that arrive at different times or are of short duration set or reset each latch. Such inputs normally come from asynchronous data sources. Once the input has gone Low (or High), the output stays in the corresponding state until reset. Again, the Enable can be used to frame data entry into the latch.

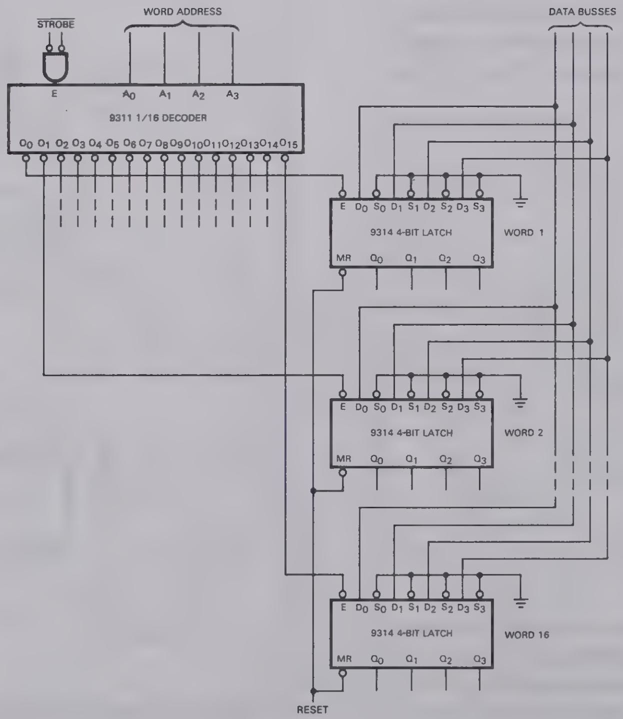
9314 4-BIT LATCH APPLICATIONS

COUNT AND HOLD DISPLAY SYSTEM


COUNT ENABLE
CLOCK
CEP
CET
CP
PE P0 P1 P2 P3
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
LATCH ENABLE
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3
A0 A1 A2 A3 LT RBI
9317 7-SEGMENT DECODER
RBO a b c d e f g
TO THE DISPLAY

A primary application of the 9314 is in count and hold display systems. The 9314 is used as a D type storage latch with the 9310 decade counter and the 9317 7-segment decoder/driver. Counter data is strobed by the Enable line and held for display until the next strobe.

LATCH SELECTION


STROBE
WORD ADDRESS
DATA BUSSES
E
A0 A1 A2 A3
9311 1/16 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
E D0 S0 D1 S1 D2 S2 D3 S3
9314 4-BIT LATCH
MR Q0 Q1 Q2 Q3
WORD 1
WORD 2
WORD 16
RESET

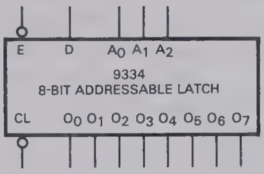
The 9314 Enable and Reset polarities are compatible with MSI decoder outputs allowing the decoders to be used in selection schemes. In this 16-latch array, the information on the parallel data busses is stored in the latches selected by a 9311 decoder. The word address inputs should be present before the decoder is strobed. Since the 9314 Enable requires only one unit load, 10 9314s representing 40 bits of storage can be controlled from each decoder output.

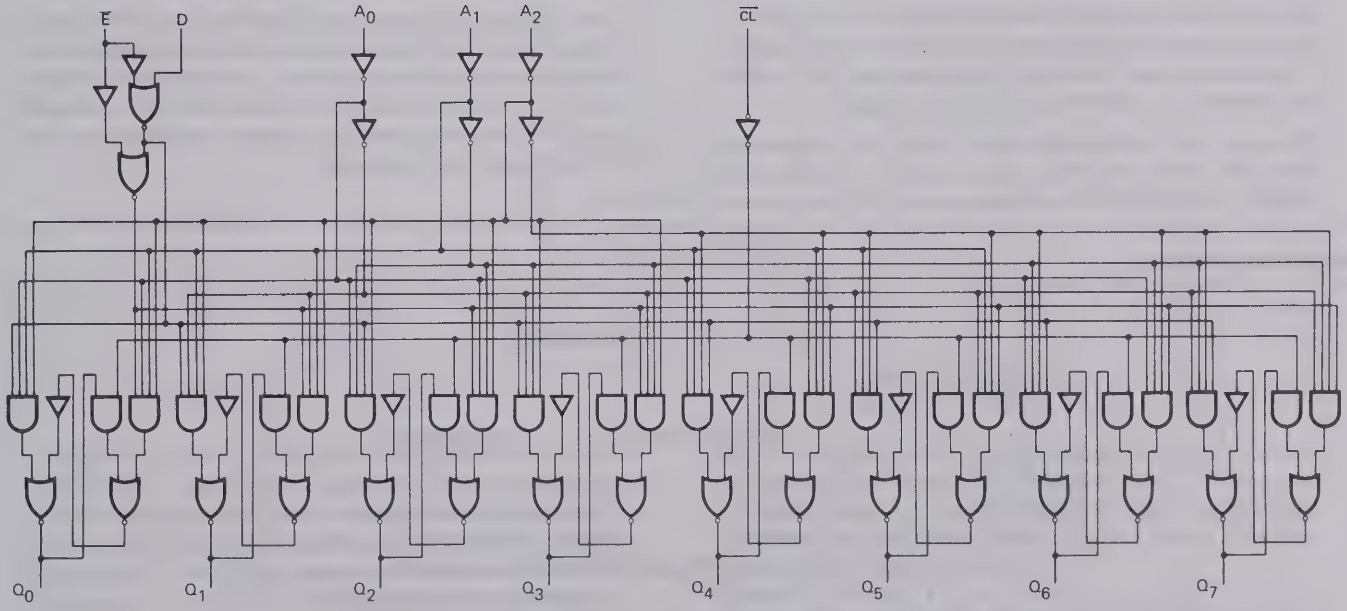
9334 8-BIT ADDRESSABLE LATCH

DESCRIPTION AND OPERATION

LEADS		LOADING
A_0, A_1, A_2	Address Inputs	1UL
D	Data Input	1 UL
$\bar{E}$	Enable (Active Low) Input	1.5 UL
$\overline{CL}$	Clear (Active Low) Input	1 UL
0 to 7	Parallel Latch Outputs	6 UL

CHARACTERISTICS	
TYPICAL DELAY	D to Q 17 ns
PACKAGE	16-Lead DIP or Flatpak
TYPICAL POWER DISSIPATION	250 mW





The 9334 8-bit addressable latch is designed for storage applications in high speed digital systems. It provides high density storage of eight bits with parallel outputs in a 16-lead package. The 9334 can serve as an 8-bit addressable latch and 1-of-8 decoder or demultiplexer. It is also valuable for random or sequential bit-by-bit data storage applications where all stored bits must be continuously available.

Writing into a latch uses three Address inputs and the active Low Enable. Data enters through the single data input. With the Enable Low, data can enter an addressed latch; with the Enable High, data entry is inhibited. Latch storage is controlled by the Clear input so when the clear input is Low, storage in all latches is inhibited; when High, the latches can store data.

9334 8-BIT ADDRESSABLE LATCH

MODE SELECTION

MODE SELECTION

$\overline{E}$	$\overline{CL}$	MODE
L	H	Addressable Latch
H	H	Memory
L	L	Active High 8 Channel Demultiplexer
H	L	Clear

PRESENT OUTPUT STATES

$\overline{CL}$	$\overline{E}$	D	A_0	A_1	A_2	O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7	MODE
L	H	X	X	X	X	L	L	L	L	L	L	L	L	CLEAR
L	L	L	L	L	L	L	L	L	L	L	L	L	L	DEMULTIPLEX
L	L	H	L	L	L	H	L	L	L	L	L	L	L	
L	L	L	H	L	L	L	L	L	L	L	L	L	L	
L	L	H	H	L	L	L	H	L	L	L	L	L	L	
L	L	H	H	H	H	L	L	L	L	L	L	L	H	
H	H	X	X	X	X	Q_{N-1}	→	→	→	→	→	→	→	MEMORY
H	L	L	L	L	L	L	Q_{N-1}	Q_{N-1}	Q_{N-1}	→	→	→	→	ADDRESSABLE LATCH
H	L	H	L	L	L	H	Q_{N-1}	Q_{N-1}	→	→	→	→	→	
H	L	L	H	L	L	Q_{N-1}	L	Q_{N-1}	→	→	→	→	→	
H	L	H	H	L	L	Q_{N-1}	H	Q_{N-1}	→	→	→	→	→	
H	L	L	H	H	H	Q_{N-1}	→	→	→	→	→	Q_{N-1}	L	
H	L	H	H	H	H	Q_{N-1}	→	→	→	→	→	Q_{N-1}	H	

The 9334 can be used as an addressable latch and a 1-of-8 demultiplexer or decoder with active High outputs. To perform these functions, the 9334 has four modes of operation, as shown in the tables above.

When in the addressable latch mode, the addressed latch will follow the data input and all non-addressed latches remain in their previous state. In the memory mode, all data is stored and all latches remain in their previous state unaffected by the Address or Data inputs. While in the demultiplexing mode, the addressed output follows the state of the D input with all other outputs Low. In the clear mode, all outputs are held Low and are unaffected by the inputs.

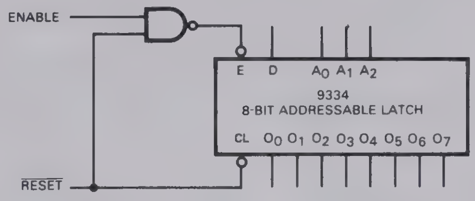
OPERATING NOTES

PRESENT STATE		NEXT STATE		RESTRICTIONS	COMMENTS
$\overline{E}$	$\overline{CL}$	$\overline{E}$	$\overline{CL}$		
H	H	L	L	None	Memory to Demultiplex
L	H	H	L	None	Addressable Latch to Clear
H	L	L	H	$A_0A_1A_2$ Stable	Clear to Addressable Latch
L	L	H	H	E Overlap C	Demultiplex to Memory

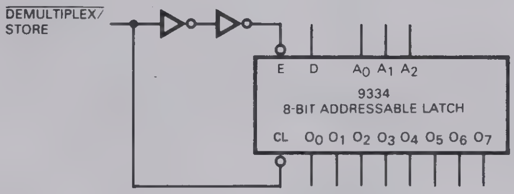
When operating the 9334 as an addressable latch, changing more than one bit of the address could impose a transient wrong address loading data into the wrong latch. Therefore, this should be done only while in the memory mode, i.e. while disabled. The operating mode is varied by changing the Enable and Clear inputs. Operating mode changes requiring simultaneous Enable and Clear transitions are accomplished as shown in the transition table. Entry into either the clear or demultiplex modes is not critical, since storage is inhibited in both cases. When entering the addressable latch mode from the clear mode, the Address should remain stable. When entering the memory mode from the demultiplex mode, the Enable should be deactivated slightly later than the Clear.

9334 8-BIT ADDRESSABLE LATCH

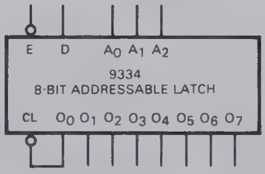
ALTERNATE MODE SELECTION SCHEMES


ENABLE
RESET
E
D
A_0 A_1 A_2
9334
8-BIT ADDRESSABLE LATCH
CL
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7

Addressable Latch with Overriding Master Reset


DEMULTIPLEX/
STORE
E
D
A_0 A_1 A_2
9334
8-BIT ADDRESSABLE LATCH
CL
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7

Demultiplexer With Storage


E
D
A_0 A_1 A_2
9334
8-BIT ADDRESSABLE LATCH
CL
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7

Addressable Clear or Reset

All four modes of operation of the 9334 may be required in some applications. However, many applications require only two or three modes. For example, when using the 9334 as a decoder/demultiplexer or addressable latch, the Clear line is grounded or held High respectively. Other combinations are shown above.

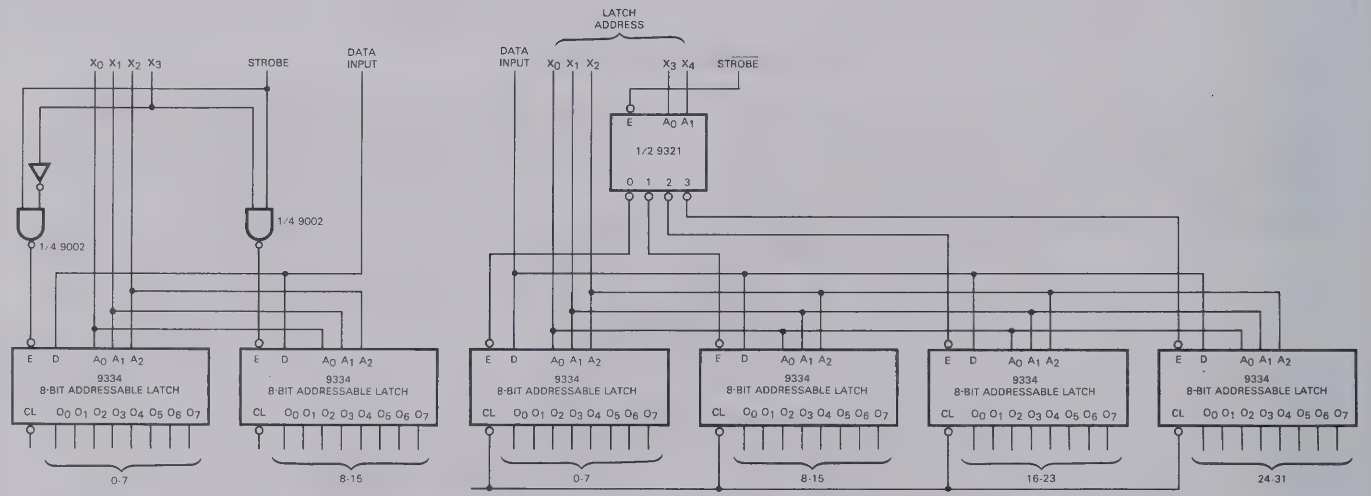
In most latch applications it is sufficient to bring the Clear line Low to reset the 9334 latches. However, an active Low overriding Master Reset can be incorporated on the addressable latch as shown to override all inputs and force all outputs to zero. When the Reset line is Low, the 9334 is forced into the clear mode. When the Reset is removed, the Enable places the latch in either the addressable latch or memory mode.

A decoder/demultiplexer with storage is also shown. With the control line Low, the selected output follows the Data input. When the line goes High, the data on the selected output is stored and allows the address to change or be removed. The inverters insure that Enable slightly overlaps Clear.

To implement an addressable reset, one latch location is reserved to reset the 9334 by connecting this latch output to the Clear input. When this location is addressed and the 9334 enabled, a zero on the D Input resets all the latches. To resume normal operation, the latch is addressed and the D Input is changed from Low to High, removing the Low level on the latch output and Clear input. This places the 9334 in the addressable latch mode, where the Enable can be removed and other latches addressed.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

ADDRESSABLE LATCH EXPANSION



X0 X1 X2 X3
STROBE
DATA INPUT
1/4 9002
1/4 9002
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
0-7
8-15
LATCH ADDRESS
DATA INPUT
X0 X1 X2
X3 X4
STROBE
1/2 9321
E A0 A1
0 1 2 3
0-7
8-15
16-23
24-31

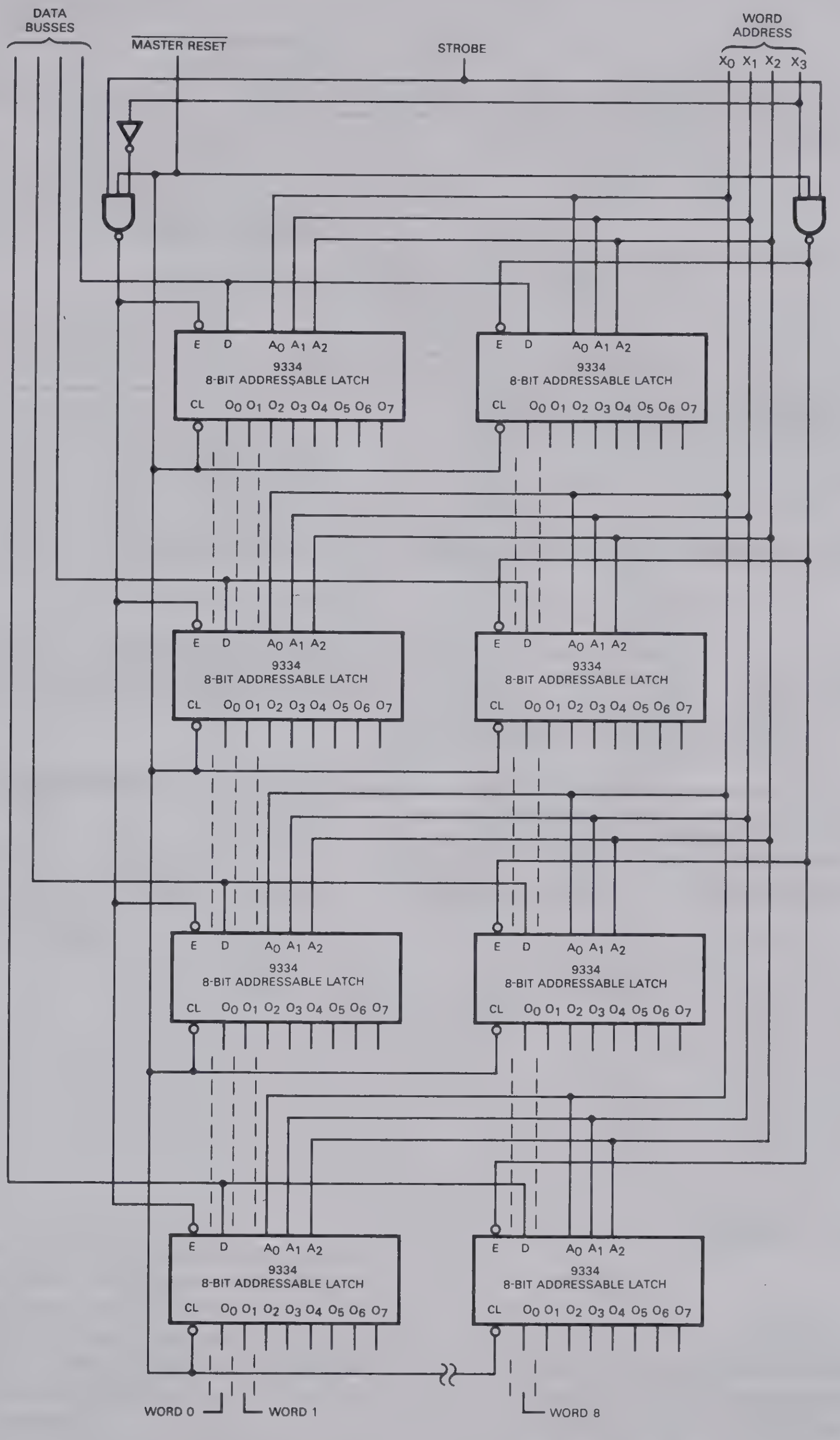
The 9334 is easily expanded to form larger storage arrays. Additional 9344s with expanded addressing can form addressable latch arrays of 16, 24, 32 and 64-bits or can be expanded in parallel to form memory arrays of arbitrary word lengths.

A 16-bit and a 32-bit addressable latch are shown here. The mode selection polarities of the 9334 facilitate expansion, and in both examples, the Enable is used to place the selected 9334 in the addressable latch mode and the remainder in the memory mode. The individual latch is selected by the three least significant Address inputs applied in common to all addressable latches. If 1-of-16 demultiplexing is also required, the Clear can be brought Low, the selected 9334 placed in the demultiplex mode and the others placed in the clear mode. When the Clear line is High, the Address should be stable before the Strobe is applied. An overriding Reset can be added to unconditionally place all 9334s in the clear mode and force all outputs low by gating the Reset with the X_3 Address or Strobe.

The mode selection of the 9334 allows the use of MSI decoders for expansion. In the 32-bit addressable latch, the 1-of-4 decoder activates the appropriate 9334 according to the two most significant address bits, and data enters the individual latch addressed by the three least significant address bits. Larger arrays are formed by using the 9301 1-of-10 decoder or the 9311 1-of-16 decoder and additional 9334s.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

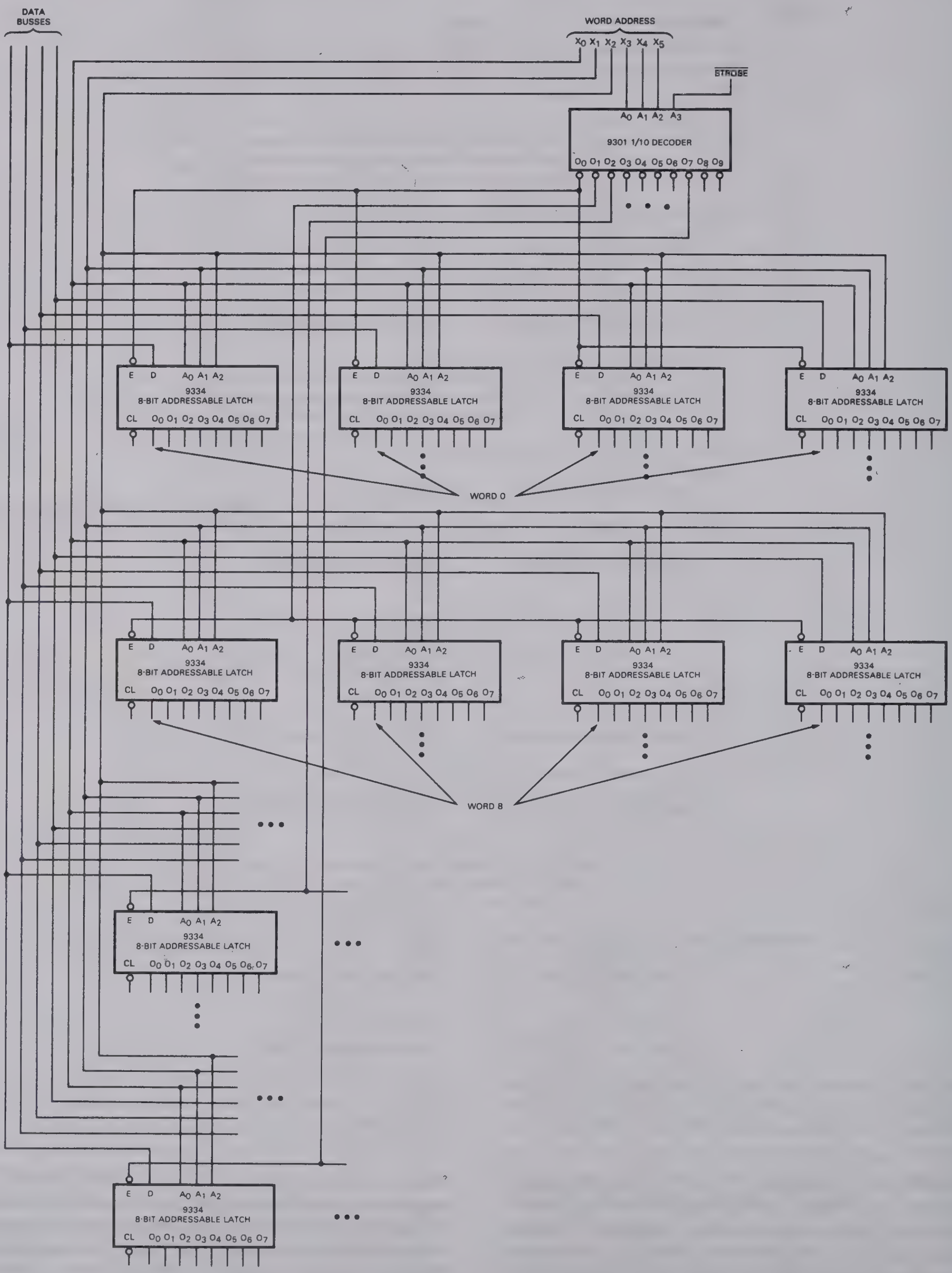
16-WORD BY 4-BIT MEMORY


DATA BUSSES
MASTER RESET
STROBE
WORD ADDRESS
X0 X1 X2 X3
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
WORD 0
WORD 1
WORD 8

The 9334 can be used as a 1-bit by 8-word memory slice, and is efficient for storing parallel data when arranged to store eight words or multiples of eight words. The 9334 provides twice as much data storage per package as other latches. Addressing 9334s in parallel obtains the desired word length and the use of 16-bit or 32-bit, etc., addressable latch arrays increases word storage capacity. The diagram for a 16-word by 4-bit memory is shown. In this memory all bits are continuously available and can be utilized to drive displays or relays, for example, in industrial control. A word is presented on the data busses, and the address ($X_3X_2X_1X_0$) determines where it will be stored. Note that the word stored in address zero appears in the zero output of the left column of 9334s. A word length of eight bits would require eight additional 9334s driven in parallel from the four Address lines ($X_0X_1X_2X_3$). The overriding $\overline{\text{Master Reset}}$ unconditionally resets the memory.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

64-WORD BY 4-BIT MEMORY

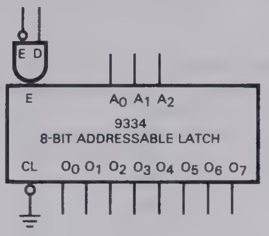

DATA BUSSES
WORD ADDRESS
X0 X1 X2 X3 X4 X5
STROBE
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
WORD 0
WORD 8

A 64-word by 4-bit memory scheme using the 9301 decoder is shown here. Data is entered while the $\overline{\text{Strobe}}$ is Low. The word stored in address zero appears at the zero outputs on the top row of 9334s. The word stored in address one appears at the one outputs of the top row of 9334s, and so on.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

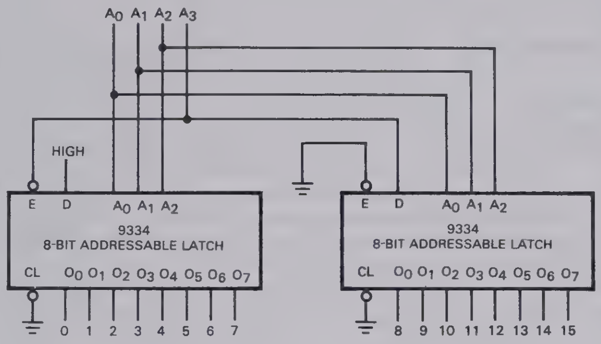
DEMULTIPLEXING AND DECODING

a) **ACTIVE HIGH DEMULTIPLEXER**



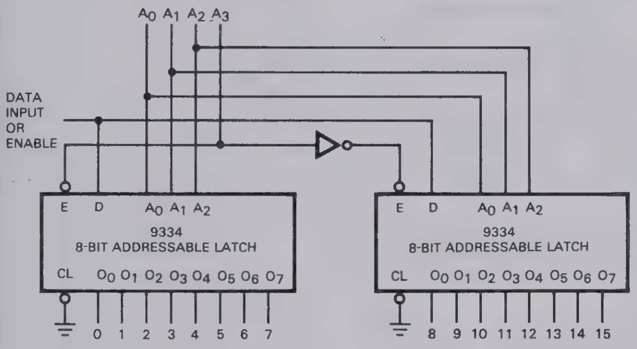
E D
E
A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL
O0 O1 O2 O3 O4 O5 O6 O7

b) **1-OF-16 DECODER**



A0 A1 A2 A3
HIGH
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
0 1 2 3 4 5 6 7
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
8 9 10 11 12 13 14 15

c) **1-OF-16 DEMULTIPLEXER**



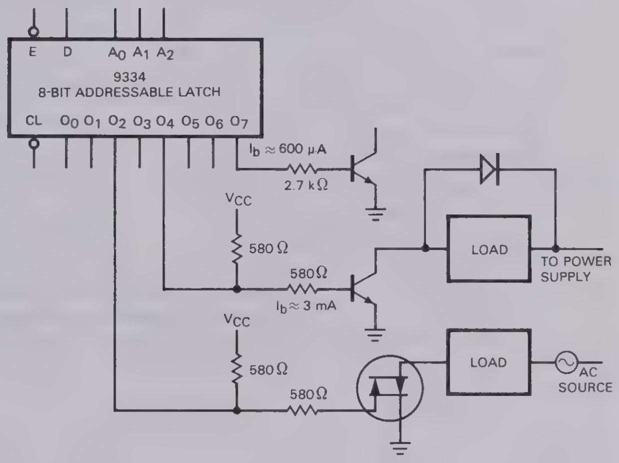
A0 A1 A2 A3
DATA
INPUT
OR
ENABLE
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
0 1 2 3 4 5 6 7
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
8 9 10 11 12 13 14 15

The 9334 can be used as a 1-of-8 active High demultiplexer or decoder. In such applications the Clear is grounded, the Enable input can be considered as an active Low Enable input and the Data input as an active High Enable input, ANDed as shown in *a* above. The Enable must be Low and the Data input must be High for an output to be High.

The 1-of-16 decoder shown is possible without additional logic by utilizing the Data and Enable inputs for device selection. A decoder Enable or Data input for demultiplexing is added by applying it to the Data inputs of both 9334s and applying A_3 and its complement to the Enable of the 9334s as illustrated.

A larger decoding array, such as a 1-of-32 decoder, can be formed by using four 9334s and an inverter. A 1-of-32 demultiplexer can be formed by using the addressable configuration in *c,* with the Clear input Low.

DRIVING DISCRETE DEVICES

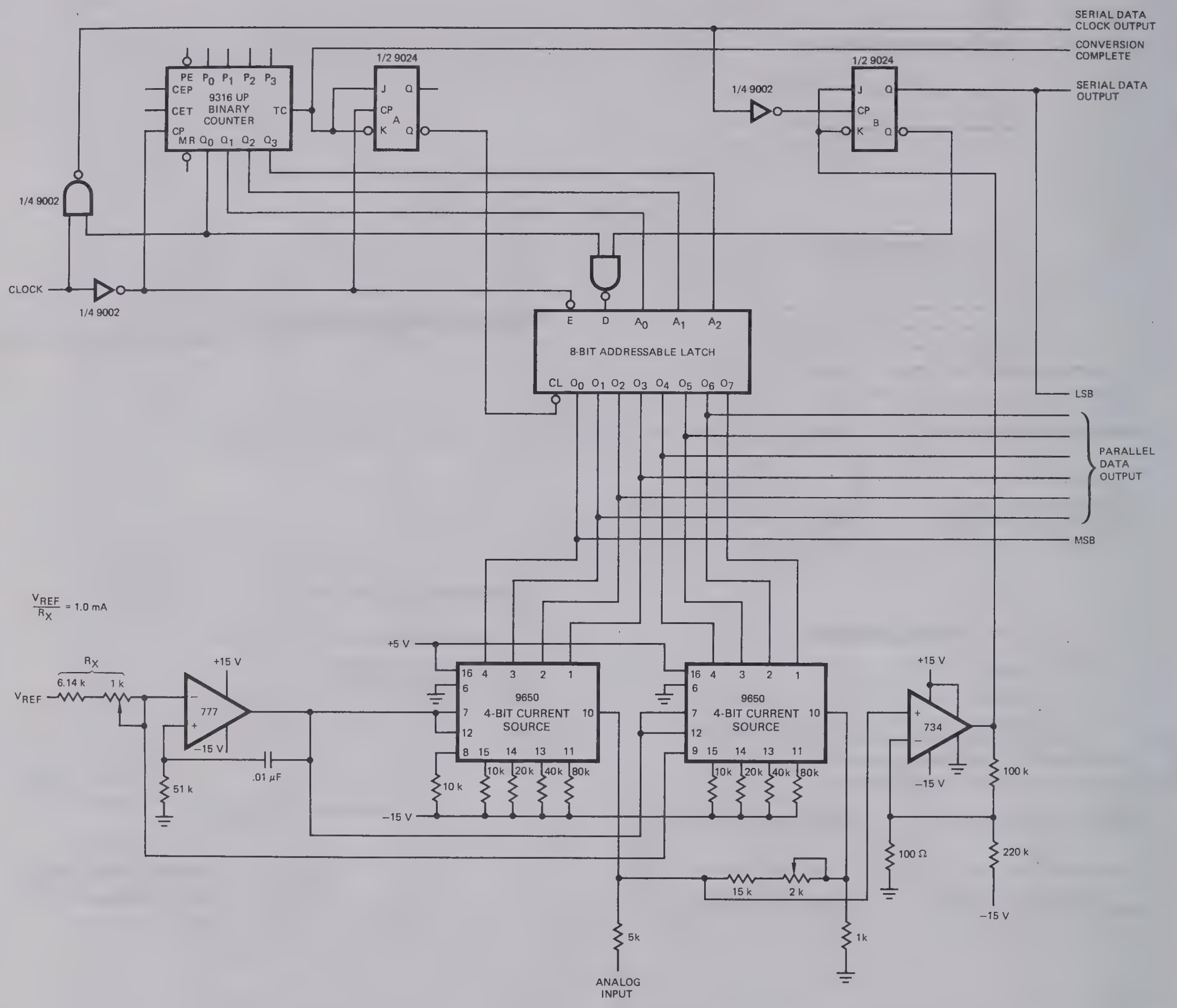


E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
Ib ≈ 600 µA
2.7 kΩ
VCC
580 Ω
580Ω
Ib ≈ 3 mA
LOAD
TO POWER SUPPLY
VCC
580 Ω
580Ω
LOAD
AC SOURCE

The 9334 can be used, for example, in industrial applications where eight or more bits of information must be stored and used to set up testing patterns to control industrial processes. In many of these applications, discrete output devices such as transistors and triacs are required. The active High outputs of the 9334 have the correct polarity to drive these discrete devices directly, so minimum interface components are required. The 9334 latch outputs have active pull ups and require current limiting in the High state when driving discrete devices. (The output high voltage must be maintained at >2.4 V for good noise immunity, since it is part of the latch feedback loop.) The pull up resistor to V_{CC} is optional and is used to supply additional base drive.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

8-BIT A/D CONVERTER


SERIAL DATA CLOCK OUTPUT
CONVERSION COMPLETE
SERIAL DATA OUTPUT
1/2 9024
1/2 9024
PE P0 P1 P2 P3
CEP
CET
CP
MR Q0 Q1 Q2 Q3
9316 UP BINARY COUNTER
TC
J Q CP A K Q
J Q CP B K Q
1/4 9002
1/4 9002
1/4 9002
CLOCK
E D A0 A1 A2
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
LSB
PARALLEL DATA OUTPUT
MSB
VREF/RX = 1.0 mA
RX
6.14 k
1 k
VREF
+15 V
777
−15 V
51 k
.01 μF
+5 V
−15 V
9650 4-BIT CURRENT SOURCE
9650 4-BIT CURRENT SOURCE
16 4 3 2 1
6
7
12
8 15 14 13 11
9 15 14 13 11
10
10k 20k 40k 80k
10 k
+15 V
734
−15 V
100 k
220 k
100 Ω
15 k
2 k
5k
1k
−15 V
ANALOG INPUT

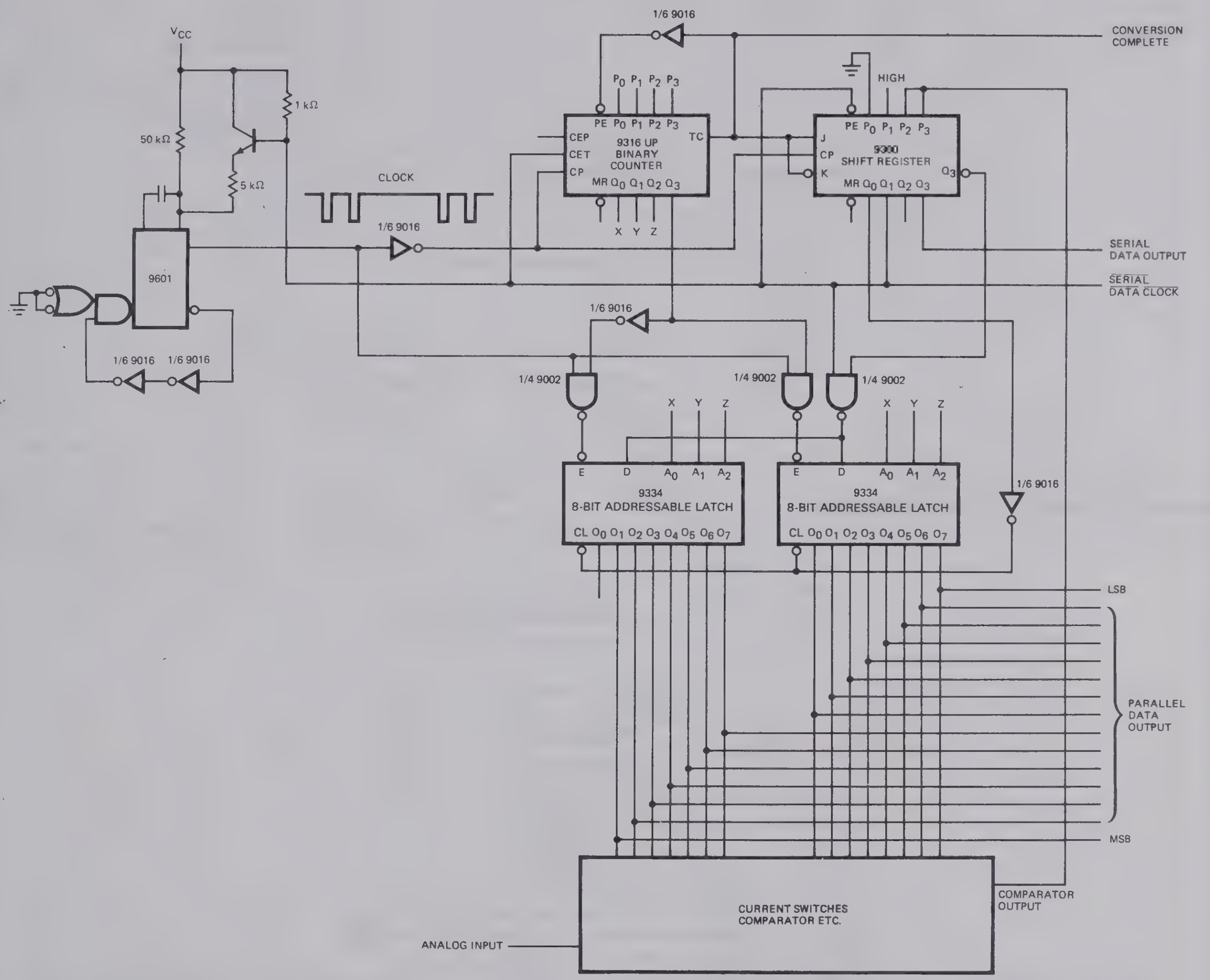
This circuit is an 8-bit successive approximation analog to digital converter. A complete conversion cycle occurs every 16 clock periods with the eight bits generated successively from most significant to least significant bit. During each even clock period the appropriate latch is forced to a one, and during the subsequent odd clock period this latch is conditionally reset. After a one is inserted into a latch the analog equivalent of the 9334 contents is compared with the input voltage. If the 9334 analog equivalent is greater, then the latch is reset; thus, a digital representation is obtained by building up an analog voltage that is compared with the analog input voltage.

After 15 clock periods, the conversion is complete and during the next clock period, Terminal Count (TC) can be used to strobe the parallel data from the converter. Serial output data is available and can be clocked from the converter by means of an output clock operating at one half the input clock frequency.

Besides the counter and addressable latch, two flip-flops provide delay and isolation. Flip-flop B permits only one logic decision every two clock periods to prevent possible oscillation between the digital and analog circuits and provides the serial output. The TC output is delayed one clock period by flip-flop A and used to clear the 9334 initializing the conversion cycle. The clock frequency, discussed in more detail later, is determined primarily by the fact that the clock must be High longer than the settling time of the linear circuits.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

PROGRAMMABLE A/D CONVERTER LOGIC


1/6 9016
VCC
CONVERSION COMPLETE
1 kΩ
50 kΩ
5 kΩ
P0 P1 P2 P3
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
X Y Z
HIGH
J
CP
K
9300 SHIFT REGISTER
Q3
CLOCK
9601
1/6 9016
SERIAL DATA OUTPUT
SERIAL DATA CLOCK
1/6 9016
1/6 9016 1/6 9016
1/4 9002
1/4 9002
1/4 9002
X Y Z
X Y Z
E D A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
1/6 9016
LSB
PARALLEL DATA OUTPUT
MSB
CURRENT SWITCHES COMPARATOR ETC.
COMPARATOR OUTPUT
ANALOG INPUT

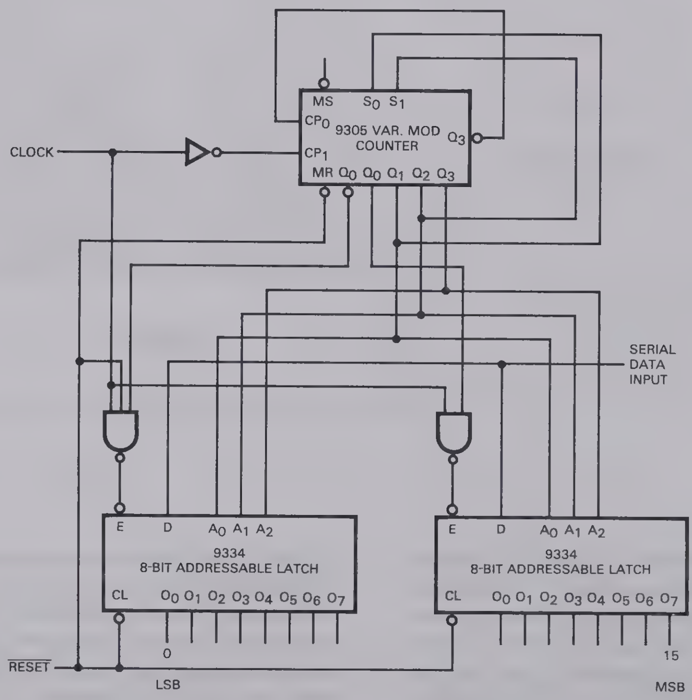
Another A/D converter capable of handling longer words and still requiring only six packages is shown in this diagram. The mode of operation is identical with the 8-bit converter but additional parts are required to handle the longer words. The converter logic can be easily programmed for 9, 10, 11, 12, 13, 14 or 15 bits by applying the appropriate code to the $P_0P_1P_2P_3$ inputs. As with the other converter, both serial and parallel output data is available.

With this converter logic the two flip-flops in the 8-bit converter are replaced with a 9300 shift register and the addressable latch is expanded to 16 bits. The shift register performs the same functions as flip-flop A & B and serves as the least significant toggle stage of the counter.

To obtain maximum speed, the clock for these converters should be asymmetrical with the clock High a majority of the time to allow the linear circuits to settle. Also, since the analog circuitry tends to be slower than the digital logic, a relatively long period should be provided after setting a one to the latch (occuring during an even clock) while a shorter period is sufficient when conditionally resetting the latch, (occuring during an odd clock). A clock generator providing these unequal clock periods is shown in the next figure. The synchronization between the converter logic and the clock generator is provided by connecting the first toggle stage of the counter to this multivibrator thus alternately producing short and long clock pulses coinciding with the state of the toggle stage.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

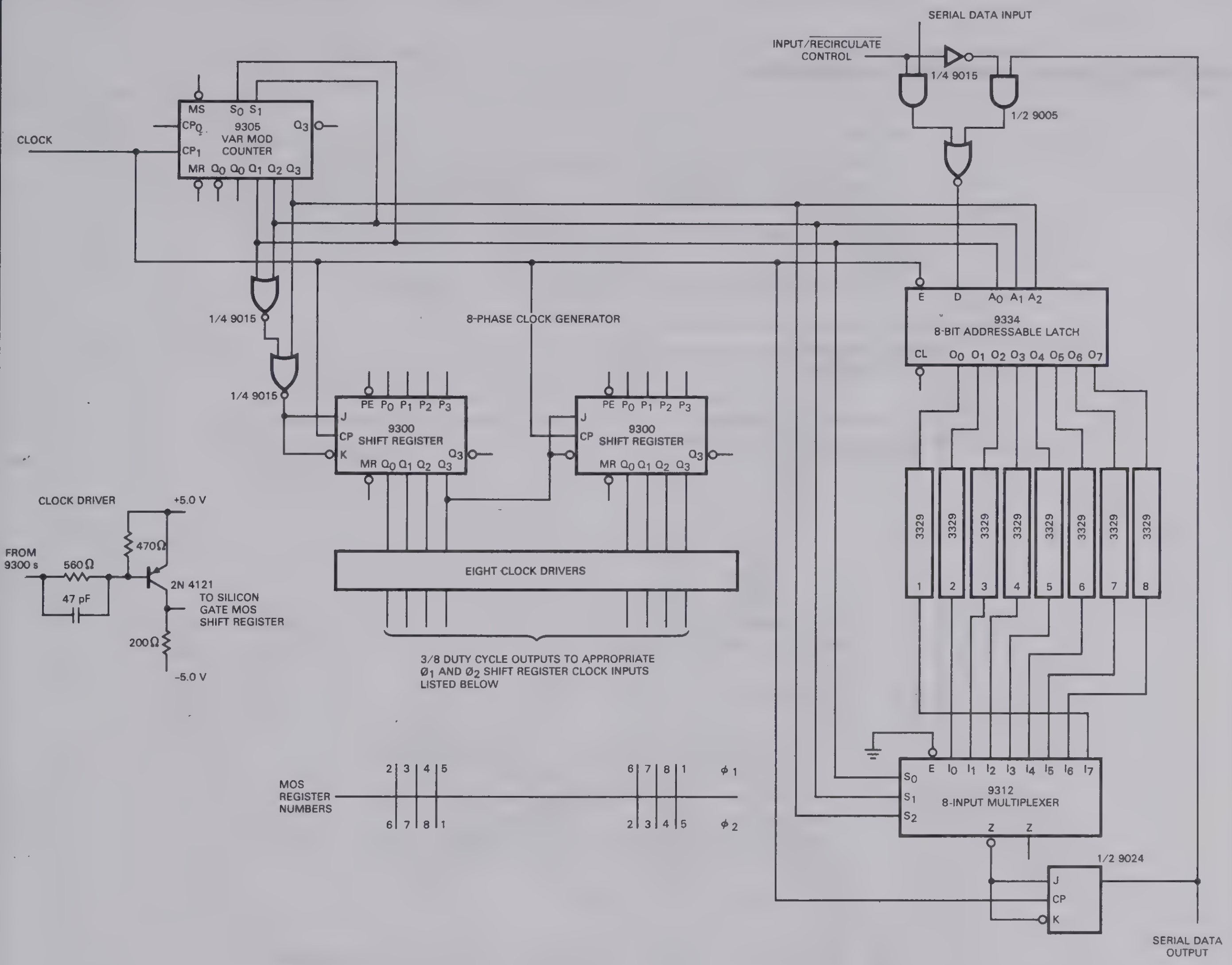
SERIAL-TO-PARALLEL CONVERSIONS


MS
S0 S1
CP0
9305 VAR. MOD COUNTER
Q3
CLOCK
CP1
MR Q0 Q0 Q1 Q2 Q3
SERIAL DATA INPUT
E
D
A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL
O0 O1 O2 O3 O4 O5 O6 O7
E
D
A0 A1 A2
9334
8-BIT ADDRESSABLE LATCH
CL
O0 O1 O2 O3 O4 O5 O6 O7
RESET
0
LSB
15
MSB

This 16-bit serial-to-parallel converter uses a binary counter (9305) to supply the addressing for a 16-bit addressable latch. The addressable latch strobe is connected to the clock so that as a new address is selected, the latch is disabled. The most significant address bit is supplied from Q_0 and the three other address bits from Q_1, Q_2, Q_3 respectively. When using the 9334 in serial to parallel converters, the parallel data bits do not change after each clock period as in a shift register. This characteristic allows the parallel data to be accepted by slower speed, lower cost and higher density devices such as MOS shift registers, as in the 4096-bit serial memory described next.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

4096-BIT, 15 MHz SERIAL MEMORY

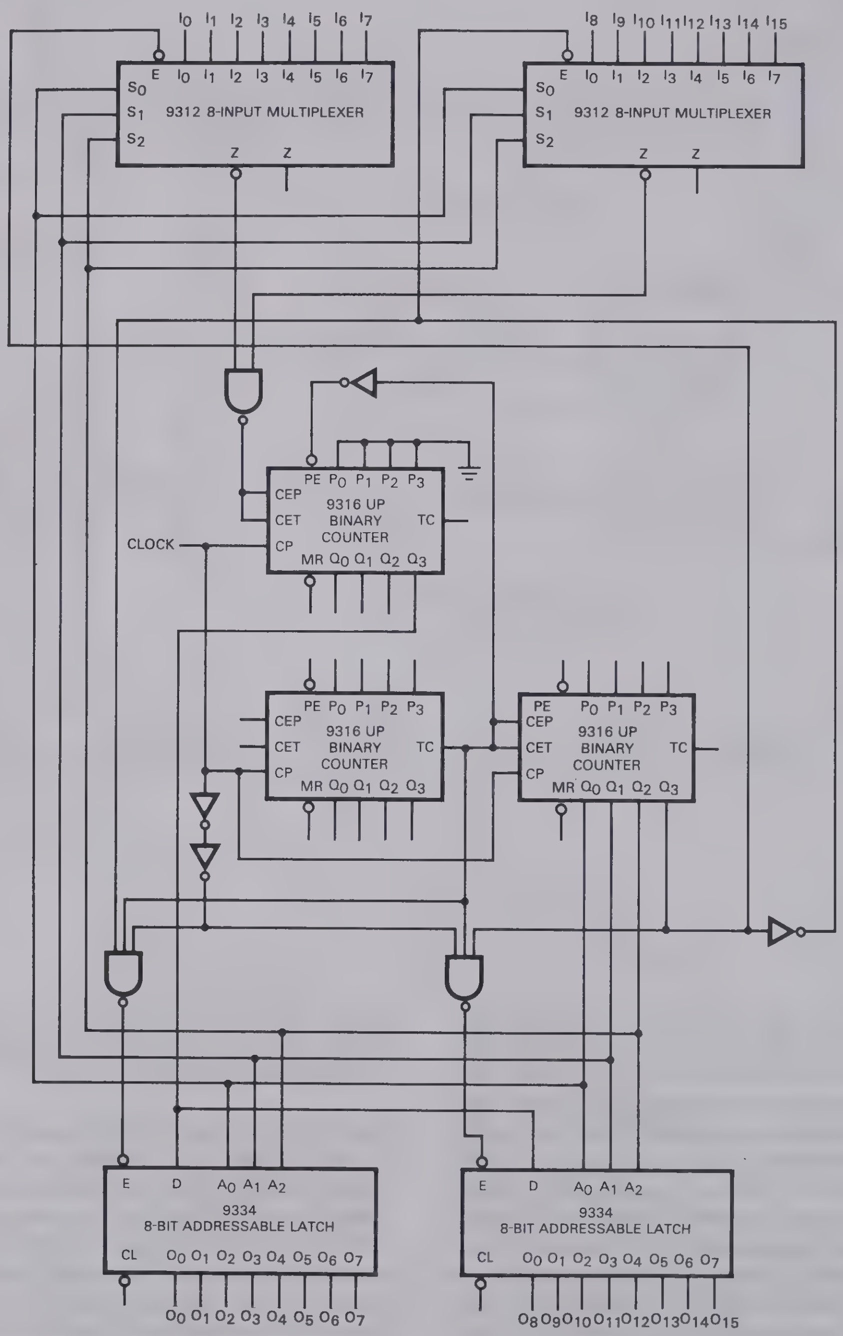

SERIAL DATA INPUT
INPUT/RECIRCULATE CONTROL
1/4 9015
1/2 9005
CLOCK
MS S0 S1
CP0
9305 VAR MOD COUNTER
CP1
Q3
MR Q0 Q0 Q1 Q2 Q3
1/4 9015
1/4 9015
8-PHASE CLOCK GENERATOR
E D A0 A1 A2
9334 8-BIT ADDRESSABLE LATCH
CL O0 O1 O2 O3 O4 O5 O6 O7
PE P0 P1 P2 P3
J
CP
K
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
PE P0 P1 P2 P3
J
CP
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
3329
1 2 3 4 5 6 7 8
CLOCK DRIVER
+5.0 V
470Ω
FROM 9300 s
560Ω
47 pF
2N 4121
TO SILICON GATE MOS SHIFT REGISTER
200Ω
-5.0 V
EIGHT CLOCK DRIVERS
3/8 DUTY CYCLE OUTPUTS TO APPROPRIATE Ø1 AND Ø2 SHIFT REGISTER CLOCK INPUTS LISTED BELOW
MOS REGISTER NUMBERS
2 3 4 5
6 7 8 1
6 7 8 1
2 3 4 5
ϕ1
ϕ2
S0 E I0 I1 I2 I3 I4 I5 I6 I7
S1
S2
9312 8-INPUT MULTIPLEXER
Z Z
1/2 9024
J
CP
K
SERIAL DATA OUTPUT

This serial memory is the equivalent of a long, high speed shift register. It combines the speed of TTL with the high density, low power and low cost of MOS. This memory can be used to simulate disk, drum or shift register memories in such applications as the refresh buffer in a CRT display, the buffer register in a printer or other computer peripheral, or in a recirculating random access memory. In this scheme, serial input data is converted to parallel, stored in MOS shift registers, and reconverted to a serial output. The system takes advantage of the fact that parallel data on each 9334 output is present for nearly eight clock cycles and consequently can be accepted by slower (2 MHz) but higher density MOS shift registers. The incoming high speed serial data bits are routed sequentially to the latches, the first data bit to latch 0, the second bit to latch 1, and so on. The MOS shift registers are driven by an 8-phase clock generator, staggered to match the data appearing on the 9334 outputs. The first MOS shift register accepts data from latch 0 after the new data appears and is ready to accept the next bit of information eight clock periods later. The clock to the next MOS register are delayed one clock period so that the register accepts data from latch 1.

Three successive states of the eight of the 3-bit address counter (9305) are decoded into the two 9300 shift registers. This produces staggered signals of 37% duty cycles at each shift register output which are fed through clock drivers to the appropriate ϕ_1 and ϕ_2 shift register Clock inputs. The chart underneath the 9300s shows the ϕ_1 and ϕ_2 connections for the registers. The order in which data from the MOS registers is reserialized is shifted one bit and resynchronized by a flip-flop. This insures high speed performance and preserves the memory length. The memory capacity of 4096 bits can be increased by cascading the MOS shift registers. For example, an additional 3329 can be connected in series with each existing 3329 to produce an 8192-bit memory. For more parallel channels, the same counter and clock generator can be used to drive the additional 9334, 9312 and MOS registers required.

9334 8-BIT ADDRESSABLE LATCH APPLICATIONS

NOISE AND CONTACT BOUNCE SUPPRESSION



I0 I1 I2 I3 I4 I5 I6 I7
I8 I9 I10 I11 I12 I13 I14 I15
E
S0
S1
S2
9312 8-INPUT MULTIPLEXER
Z
Z
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
D
A0 A1 A2
9334 8-BIT ADDRESSABLE LATCH
CL
O0 O1 O2 O3 O4 O5 O6 O7
O8 O9 O10 O11 O12 O13 O14 O15

A digital method for masking noise and bounce on inputs to a digital system is shown. This circuit performs this masking function for 16 inputs.

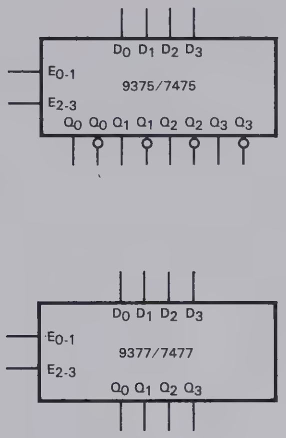
Each input is sequentially sampled for 16 clock periods. When an input is High for eight or more clock periods, the output corresponding to that input is set High; otherwise the output is set Low. All 16 inputs are sampled in this manner and all outputs are updated as a result.

Assume a 10 kHz clock rate for this application. Each input is sampled for 1.6 ms every 25.6 ms. Thus, to affect the output, a noise pulse must reverse the input for at least 800 μs and a contact bounce must last longer than 25.6 ms.

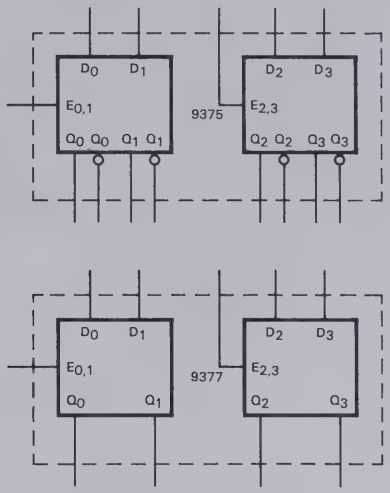
9375/5475, 7475; 9377/5477, 7477 4-BIT LATCHES

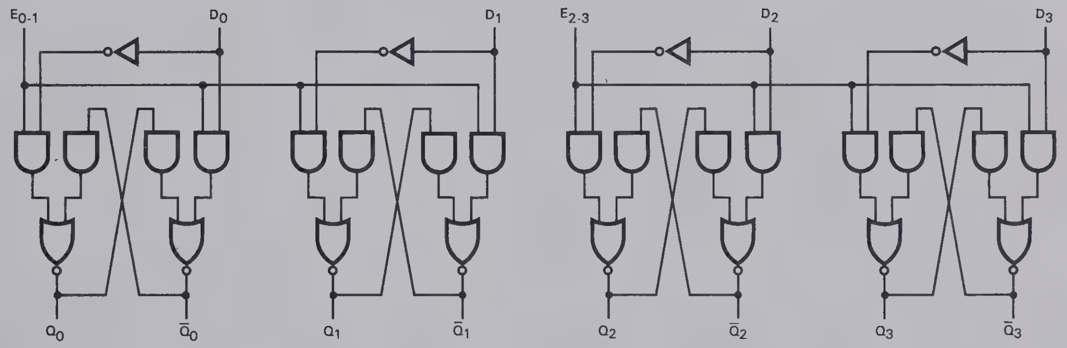
DESCRIPTION AND OPERATION

STANDARD LOGIC SYMBOLS


D0 D1 D2 D3
E0-1
9375/7475
E2-3
Q0 Q̄0 Q1 Q̄1 Q2 Q̄2 Q3 Q̄3
D0 D1 D2 D3
E0-1
9377/7477
E2-3
Q0 Q1 Q2 Q3

ALTERNATE LOGIC SYMBOLS


D0 D1
E0,1
Q0 Q̄0 Q1 Q̄1
9375
D2 D3
E2,3
Q2 Q̄2 Q3 Q̄3
D0 D1
E0,1
Q0 Q1
9377
D2 D3
E2,3
Q2 Q3


E0-1
D0
D1
E2-3
D2
D3
Q0
Q̄0
Q1
Q̄1
Q2
Q̄2
Q3
Q̄3

LEADS		LOADING
D_0, D_1, D_2, D_3	Data Inputs	2 UL
E_{0-1}	Clock Input Latches 0 & 1	4 UL
E_{2-3}	Clock Input Latches 2 & 3	4 UL
Q_0, Q_1, Q_2, Q_3	Latch Outputs	10 UL
$\overline{Q}_0, \overline{Q}_1, \overline{Q}_2, \overline{Q}_3$	Complementary Latch Outputs	10 UL

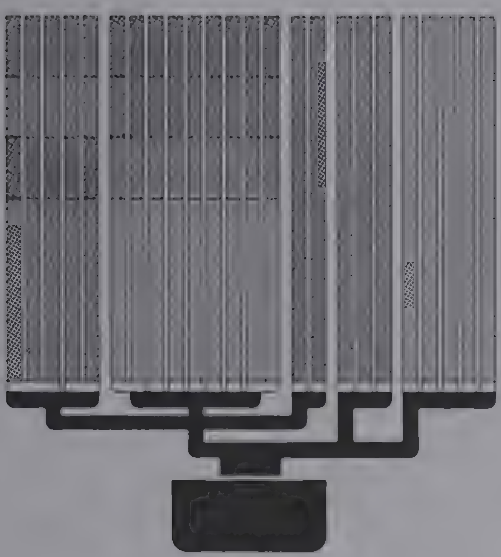
E	D	Q_N	OPERATION
H	L	L	DATA ENTRY
H	H	H	DATA ENTRY
L	X	Q_{N-1}	HOLD

The 9375 and 9377 4-bit latches are suitable for temporary storage of digital information typically between processing units and input/output on indicator units. Both devices are compatible with all other members of the TTL/MSI family.

The logic symbols, alternate logic symbols and logic diagram for the 9375 and 9377 are shown above.

The 9375 and 9377 feature two separate 2-bit D latch sections each with an active High enable. The 9375 has complementary outputs; the 9377 does not. The alternate logic symbols present the active Low clock inputs as active High Enables, consistent with the nomenclature traditionally used for single-rank storage devices (latches) in the Fairchild TTL family.

Information present at a data input is transferred to the output when the Enable (or Clock) is High. The output will follow the data input as long as the Enable remains High. When the Enable is Low, the information that was present at the data input at the time of the High-to-Low transition is retained until the Enable is permitted to go High again.



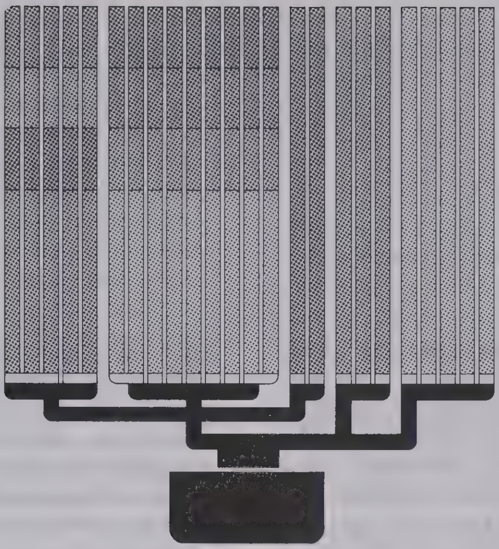
Bipolar Memories

Memory Organization
Addressing Techniques
General Timing Considerations
RAMs
Word Expansion
256 By 8-Bit Buffer Memory System
LIFO Push Down Memory Stack
ROMs
Expanding ROMs
Changing ROM Format

7
8
9
10
11
12
13
14
15

MEMORY SELECTION GUIDE

Function	Device	Size Words x Bits	Chip Select	Address Access Time, ns		Required Write Pulse Width		Power Dissipation/Bit, mW		Input Loading
				Typical	Max	Worst Case	Typical	Typical	Max	
Read/Write	93403	16 x 4	1	45	60	45	30	7.0	9.0	1.0 UL
	93410	256 x 1	3	45	60	30	25	1.8	2.8	0.5 UL
	93410A	256 x 1	3	35	45	30	20	1.8	2.8	0.5 UL
	93415	1024 x 1	1	60	90	55	30	0.4	0.7	0.25 UL
Read Only	93434	32 x 8	1	32	50	NA	NA	1.3	1.7	1.0 UL
	93406	256 x 4	2	30	50	NA	NA	0.5	0.67	0.5 UL



BIPOLAR MEMORIES

INTRODUCTION

Bipolar memories fall into two very different categories: random access read/write memories (RAMs) and read only memories (ROMs).

A RAM is an array of latches with a common addressing structure for both reading and writing. A Write Enable input defines the mode of operation. In the write mode, the information at the Data input is written into the latch selected by the address. In the read mode, the contents of the selected latch is fed to the Data output.

All semiconductor memories have non-destructive readout as opposed to the destructive readout of most magnetic core memories. Bipolar RAM operation is static, i.e., the information is stored in bistable transistor cells (latches) and requires no refreshing such as required in some popular MOS RAMs using capacitor storage. Data storage in all semiconductor read/write memories is volatile; data can only be stored as long as power is uninterrupted.

In a read only memory (ROM), the data content is fixed, normally by a unique metallization of the chip. Addressing is similar to that of a RAM; the operation is static and the readout is obviously non-destructive. A ROM offers non-volatile storage; data is retained indefinitely even when power is shut off.

Today, bipolar memories are an integral part of a large number of digital equipment designs. In six years, density has increased by a factor of 64, with a corresponding decrease in power per bit without penalizing speed. Fairchild is currently producing 1024 bit RAMs with 500 mW typical power and 60 ns typical access time.

MEMORY ORGANIZATION

Memory subsystems are generally identified by number of words, number of bits and function. For example, a 1024 x 16 RAM is a random access read/write memory containing 1024 words of 16 bits each. Semiconductor memory device organizations follow the same rule. Since the advent of LSI allowing densities of hundreds of gates on a chip, most memory devices contain address decoders, output sensing, and various control and buffer/driver functions in addition to the array of storage cells. High density RAM devices tend to be organized n words by one bit to optimize lead usage. ROM devices tend toward n words by four or eight bits to reduce cost of truth table changes.

ADDRESSING TECHNIQUES

Addressing (word selection) in a semiconductor memory subsystem consists of two parts. First, a given device or group of devices must be selected, second, a given location in a device or group of devices must be selected. Device selection may be accomplished by linear select using a binary-to-n decoder feeding the chip select function on n chips, or by coincident select using two binary-to-$\sqrt{n}$ decoders and two chip selects on each device. When n is large, linear select requires excessive hardware. For example, if n = 64, linear select requires four 1-of-16 decoders and a 1-of-4 decoder, or nine 1-of-8 decoders; whereas coincident selection can be accomplished with two 1-of-8 decoders with final decoding at the two input chip select gates included on the memory devices. Selection of a given location on a chip is accomplished by connecting the binary address lines directly to the chip. In summary, 64 256 x 1 RAMs in a 16K x 1 bit array using coincident selection requires 14 address lines, as follows: eight connected to 2^0 through 2^7 inputs on all chips (using necessary drivers), three feeding a 1-of-8 decoder to the CS_1 inputs, and three feeding a 1-of-8 decoder connected to the CS_2 inputs.

BIPOLAR MEMORIES

MEMORY TYPES

A number of choices must be made in selecting bipolar memory devices for a specific application. First, should TTL or ECL be used? The ever increasing need for speed in new designs may impose a propagation requirement on the memory that cannot be met with current TTL technology. This may in turn influence the selection of the logic family used for implementing the ALU or other functions associated with the memory. Currently, the fastest 256-bit RAM available is the 95410 ECL memory.

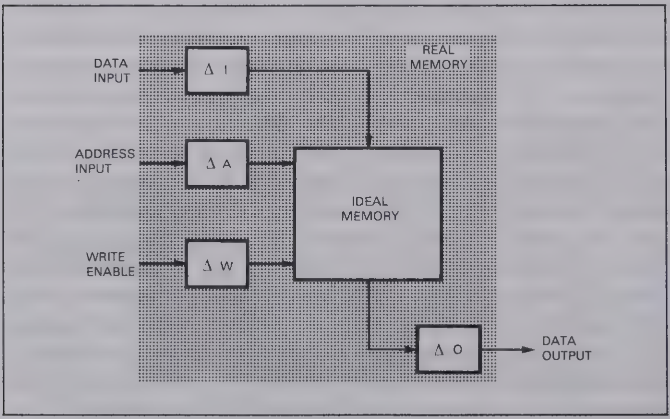
Next, should RAM or ROM be used? For data storage the choice is obviously a RAM since fast write capability is a "must". For microprogramming the decision is not so clear-cut. Since most system designs continue to evolve and change through prototyping and preproduction, RAM may be the logical choice; its higher device cost is offset by eliminating the cost of replacing ROMs as changes are made to the firmware. In this situation, RAM also offers the advantage of eliminating system downtime during firmware modification, a very significant factor in development schedules.

Numerous other decisions must be made, such as power consumption level, relationship of density and speed, and other parameters. The selection guide is provided for assistance in choosing Fairchild bipolar memories for various applications.

GENERAL TIMING CONSIDERATIONS*

A simple way of analyzing a semiconductor memory in terms of its delays is based on the assumption that it is an ideal device with three types of input and one type of output. The device may be a single storage cell, or an array of such cells. Being ideal, it has no intrinsic delays, but its external connections have individual delays — ΔI on the Data input, ΔW on the Write Enable signal, ΔA on the Address input, and ΔO on the Data output. And if the device is a large array, the external lines may be multiple, carrying whole bytes or words instead of single bits, for example.

Ideal Memory System



Each input and output in the diagram is binary — that is, it has only two stable conditions, a more negative level and a more positive one. For a given device, each delay shown has two values, one for the propagation of a negative-to-positive transition through the delay element, and one for the propagation of a positive-to-negative transition. Moreover, combinations of these input and output delays create the real device delay times. Access time, for instance, is the sum of the delays on the Address input, ΔA, and on the Data output, ΔO (since the delay through an ideal device is zero). Since ΔA and ΔO each have maximum and minimum values, their sum also has a maximum and a minimum value. Likewise, data may appear on the Input lines, and at some later time a Write Enable signal appears, making the device data setup time $\Delta I - \Delta W$.

In general, a memory system contains many devices for which the delays are different, but the maximum and minimum limits of each must be known. Moreover, when a system is intended for high volume production, the range for the devices to be used in all systems must be known to avoid the need to tune each system individually. A given pair of paths may turn out to have the maximum delay, and, for the system to operate reliably, this defines system minimum timing. Yet the designer must also remain aware of the fact that access time may be shorter than the maximum, and that the output of the memory device is unknown between $(\Delta A + \Delta O)_{min}$ and $(\Delta A + \Delta O)_{max}$.

At this point, relationships between the individual delays in the simple model must be considered. In the absolute sense, this would involve specifying the delay in every pair of paths — a rather large number of parameters. More practically, the output delay is usually important only during a read operation, so only a few measurements need be specified. But the input timing is completely under control of the system, and must be specified for every input path with respect to every other input path. Once he knows how these inputs behave in relation to each other, the designer can guarantee a reliable system.

All these delays are internal device parameters, not system parameters. But because of the widespread failure to distinguish between the two kinds of measurements, some manufacturers list a guaranteed minimum access time and others a guaranteed maximum — the maximum for the device is the minimum for the system. For clarity, this section designates all device delays with a small t and all system delays with a capital T.

There are eight important memory device delays, all some combination of the four delays in the simple model. The eight device delays are access time, read recovery time, write time, data setup time, data release time, address timing, write recovery time, and chip select delay.

---

*Reprinted in part from Electronics, October 25, 1971;
Copyright McGraw-Hill, Inc., 1971

Access time may be specified as t_{pd+} and t_{pd-}, or simply as t_{access}. It is the length of time from the appearance of a valid address on the Address input to the appearance of valid data on the Data output — and also, if specified, the time from the disappearance of the address to the disappearance of the active level. Where both are specified, system access time can sometimes be improved by taking advantages of the difference between them at the gate driven by the Data output line. Where only one is specified, it may be the larger of the two, but more often it is just t_{pd-}, the active output state being the more negative one. Since the output stage of a memory usually has open collector circuits, access involves pulling the output line from its normal positive level to the negative level if the stored data demands it.

Both maximum and minimum access times are important. The maximum access time specifies how long the system must wait for valid data after it supplies an address; the minimum time specifies how long the data remains good after the address has been removed.

The read recovery time, t_{rr} is the time required for the output of the memory to return to its normal (usually the positive) level, after the address has been removed. This delay, which is mostly caused by the sense amplifiers inside the device, must be taken into account when two successive read operations from different addresses within a single device cause the Data output line to be first Low, then High. The High level cannot be attained until after t_{rr} has elapsed. Read recovery time is basically the t_{pd+} for an open collector output. It is usually of no consequence, because it will be covered by the access time of a second read cycle.

The write time t_W is the length of time that an active level must be present on the Write Enable to guarantee successful writing in the memory. (A better name for it would be "write pulse width," but this would confuse it with a similar system parameter.) The system must provide the Write Enable signal for the full length of $t_{W(max)}$ to guarantee writing in the slowest memory devices, and if the Write Enable signal is generated by a decoder circuit, any glitches on the decoder output must be less than $t_{W(min)}$ to guarantee not writing in the fastest devices.

Because the Write Enable signal begins and ends the writing operation, it is a convenient reference for the other input delays, serving as a clock or basic timing pulse. Data setup time and the data release time for both High and Low output levels make up four parameters associated with a memory data input and symbolized as t_{dsh}, t_{dsl}, t_{drh}, and t_{drl}.

Consider the two external delays, ΔI and ΔW, shown in the ideal memory system. Since the write time is the critical parameter, let the time scale begin at the point when Write Enable becomes active — in most cases, when it undergoes a High-to-Low transition. The level on the Data line is not important at this time, because most memories accept input data up to the Low-to-High transition of the Write Enable line, just like an edge-triggered flip-flop. But when this Low-to-High transition occurs, the Data input to the ideal cell must represent the data to be stored. Thus the correct levels must be present at the input of the memory ΔI earlier. This deadline is $\Delta I - \Delta W$ before the end of the Write Enable signal at the real memory input.

Only the Low-to-High transition of the Write signal is important, but both transitions of the data are significant. These are the setup times for High and Low data.

$$t_{dsh} = \Delta I_+ - \Delta W_+$$

$$t_{dsl} = \Delta I_- - \Delta W_+$$

Suppose that these are 20 ns and 10 ns respectively. If data represented by a High level is to be stored, then it must be on the Data input lines at least 20 ns before the Write Enable signal rises — at the *real* memory, not at the *ideal* device. This High data must also remain on the Data lines long enough to get stored in the memory. Now a Low level takes 10 ns to get through the delay, and, if it appears less than 10 ns before the rise of the Write Enable signal, will not get to the ideal device before the preceding High level has been stored and the Write Enable signal is gone.

Therefore, the High level must be applied at least 20 ns, and be removed at most 10 ns, before the rise of the Write Enable signal. The 10 ns time is commonly called the data release time for the High level, or t_{drh}, but it is obviously also the data setup time for the Low level, or t_{dsl}. Hence $t_{drh} = t_{dsl}$, not only in magnitude, but in every other respect too.

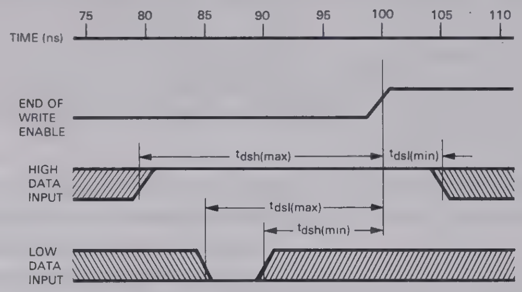
In a system containing more than one device, however, the ranges of setup and release time affect the timing of data availablability on the inputs. For example, to store data represented by a High level, the level must be applied not later than the longest t_{dsh} before the rise of Write Enable, and maintained at that level until after the beginning of the shortest t_{drh}, or equivalently the shortest t_{dsl}. Conversely, to store the Low level, it must be present before the maximum t_{dsl} and remain until after the minimum t_{dsh}.

If the delay through the data path is very short, the $\Delta I - \Delta W$ may be negative, and appear as a negative value for one or both of the minimum data setup times. That is, under certain circumstances, data that changes after the removal of the external Write Enable signal will be stored at its new value because in fact the Write Enable signal was still moving slowly through the delay when the new data arrived. Data sheets specify this as either a negative release time or a positive hold time.

The four setup and release times are really only two different quantities, which can have maximum, minimum, and sometimes negative values. Some of the difficulties in specifying them are illustrated on the following page and five methods of publishing timing specifications (A – E) are shown. The timing chart is for a memory device with a Write Enable pulse that ends (goes High) 100 ns after the reference time. The chart also shows the required inputs for High and Low data. Data can change in the shaded areas, but must be stable in the unshaded areas to guarantee writing as specified.

The first specification (A) follows the recommended method. To write High data, the Low-to-High transition must occur no later than 80 ns after the reference time, which is the maximum t_{dsh} before Write Enable rises; and the level must stay High until 105 ns after reference, a time later than the rise of Write Enable because the minimum t_{dsl} is negative. A later rise or earlier fall makes the correct storage doubtful. But to write Low data, the High-to-Low transition can occur as late as 85 ns after reference, allowing the maximum t_{dsl}, and can be reversed as early as just after 90 ns, for the minimum t_{dsh}.

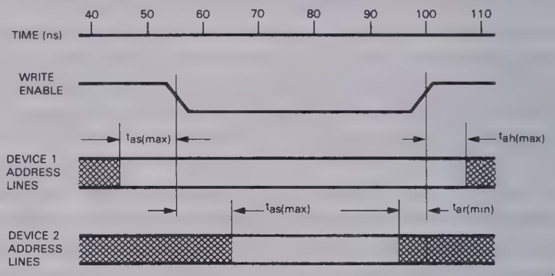
Data Timing (Write Mode)



Method	Parameter	Definition	Min	Typ	Max (ns)
A	t_{dsh} t_{dsl}	High Data Setup Time Low Data Setup Time	10 –5		20 15
B	t_s	Data Setup Time	–5		20
C	t_s t_r	Data Setup Time Data Release Time			20 –5
D	t_s t_h	Data Setup Time Data Hold Time			20 5
E	T_s T_h	System Data Setup Time System Data Hold Time	20 5		

Meaning of shading is explained on the right.

Address Timing (Write Mode)



Device	Parameter	Definition	Min	Typ	Max	Units
1	t_{as} t_{ah}	Address Setup Time Address Hold Time			10 7	ns ns
2	t_{as} t_{ar}	Address Setup Time Address Release Time	 5		–10	ns ns

Transition Time Uncertainties

TIMING DIAGRAM SYMBOL	MEANING: Forcing Functions	MEANING: Other Functions
☐	Must be Steady High or Low	Will be Steady High or Low
▧	High-to-Low Changes Permitted	Will be Changing From High to Low Sometime During Designated Interval
▨	Low-to-High Changes Permitted	Will be Changing From Low to High Sometime During Designated Interval
▩	Don't Care	State Unknown: Changing

Specification B ignores the difference between setup times for High and Low signal levels. Specification C chooses to list High data release time instead of Low data setup time. Specification D uses still another term, data hold time, which simultaneously changes the sign of the quantity and changes the minimum to a maximum, so that now the column is correct. Specification E is written from the system instead of the device point of view.

Address timing, like data timing, involves four different quantities: setup time, t_{as}, release time, t_{ar}, hold time, t_{ah}, and another quantity that is often called simply address time, or t_a. But unlike data timing, both edges of the Write Enable signal are important relative to the Address signal. In most devices, an address is always present on the Address inputs, and this must remain unchanged during the write process.

There is likely to be less difference between the times at which nominally simultaneous High and Low address bits arrive at the ideal memory than there is with data bits, because address bits usually pass through less combinational logic enroute from their source. This means the worst case fast and slow bits are usually fairly close together, and differences between High and Low address bits need not be specified. However, two parameters must be carefully considered; these are the maximum setup time relative to the leading edge of the Write Enable signal, and the minimum setup time relative to the signal trailing edge.

The first parameter is the longer of $\Delta A_+ - \Delta W_-$ and $\Delta A_- - \Delta W_-$. It specifies how much time the device with the slowest address path requires to get the address into the ideal memory ahead of the Write Enable pulse. Its symbol is $t_{as(max)}$.

The second parameter is the shorter of $\Delta A_+ - \Delta W_+$ and $\Delta A_- - \Delta W_+$. It specifies how long the fastest device takes to get an address into the ideal cell. No change in an address can be permitted until less than this time remains before the end of the Write Enable signal. Although it is actually the minimum address setup time, it is measured relative to the rising edge of Write Enable; whereas the maximum setup time is measured relative to the falling edge and deserves a different name. Since the minimum setup time is likely to be either zero or negative, it is commonly designated address hold time, or t_{ah}; where it is positive, address release time, or t_{ar} is used.

The Address Timing figure shows this for two hypothetical memories. Device 1 is typical of most semiconductor memories, but occasionally one is encountered with the characteristics of device 2. In both, a substantial difference between ΔW_+ and ΔW_- is assumed, emphasizing that the address setup and release times are not two limits of the same parameters. As the diagram and the accompanying table indicate, a release time is a minimum, but a hold time, because it has the opposite sign, is a maximum.

Although addresses for most memories are in binary form, occasionally one has some other scheme. Then the minimum address pulse width becomes important, because sometimes a nonexistent address may be presented between valid address pulses to the array. Since writing can be guaranteed only if a valid address is present long enough for the slowest memory to respond, the address time, t_a, is analogous to, and lasts about as long as, the write pulse width, t_w.

But the really important criterion for writing is that, at the ideal memory, both the address and the write pulse coincide for long enough to allow the slowest device to respond — during a maximum time, again, from the device point of view, and a minimum time from the system point of view. Either of the two pulses can start the operation when the other is present, and the trailing edge of either can end it. Generalizations are difficult but in most systems, the address brackets the write pulse, so that the write operation depends on the duration of the latter.

The write recovery time, t_{wr}, restricts the use of the Data output lines following a write operation. It is quite like the read recovery time — the sense amplifiers respond to the data being written just as they do during a read operation, and when the written data drives the output to its more negative level, the amplifiers require a short time to go positive.

An example of this behavior is found in Fairchilds 93403 4-bit by 16-word memory, with output following the input during a write operation, but inverted during a read operation. Thus, writing Low level data into the memory causes the output to be Low while the Write Enable pulse is present; but, when the Write Enable ends and the address remains unchanged, the newly written data appears as a High level at the output. Because the sense amplifier inverts the data, each stored Low level appears as a High level — but not until after the write recovery time has passed.

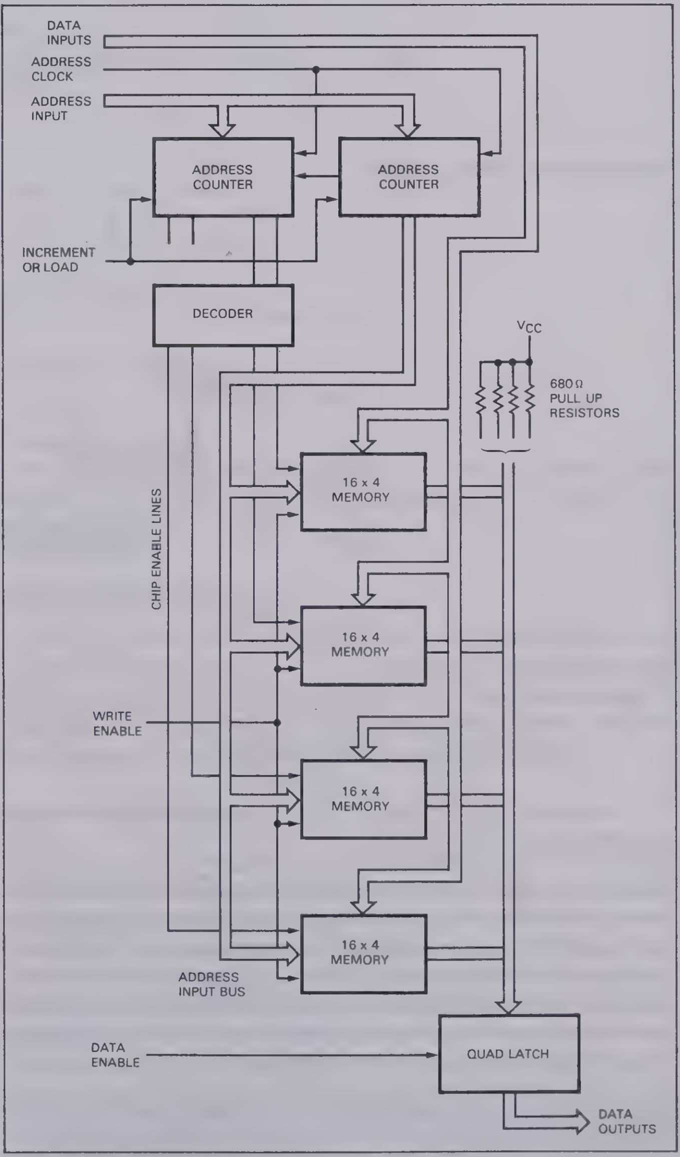
Finally, most memories have Chip Select inputs that permit address and data lines to be shared by several chips. One chip can be enabled for a read or write operation.

In some memories the Chip Select is simply part of the Address. It is decoded in the same way and is subject to the same delays. But in newer designs such as the 93410 256 x 1 TTL RAM and the 93415 1024 x 1 TTL RAM, where the Chip Select is independent of, and is considerably faster than, the Address, it has its own delay time specified. The delay from the leading edge of the Chip Select signal to the time the output becomes active is the enable time, t_e, while that from trailing edge to inactive output is disable time, t_d. For a write operation, the enable time has minimum and maximum values relative to the trailing edge of the write pulse, just as the Data input does, although in general the Chip Select signal should be active at least as long as the Write Enable signal.

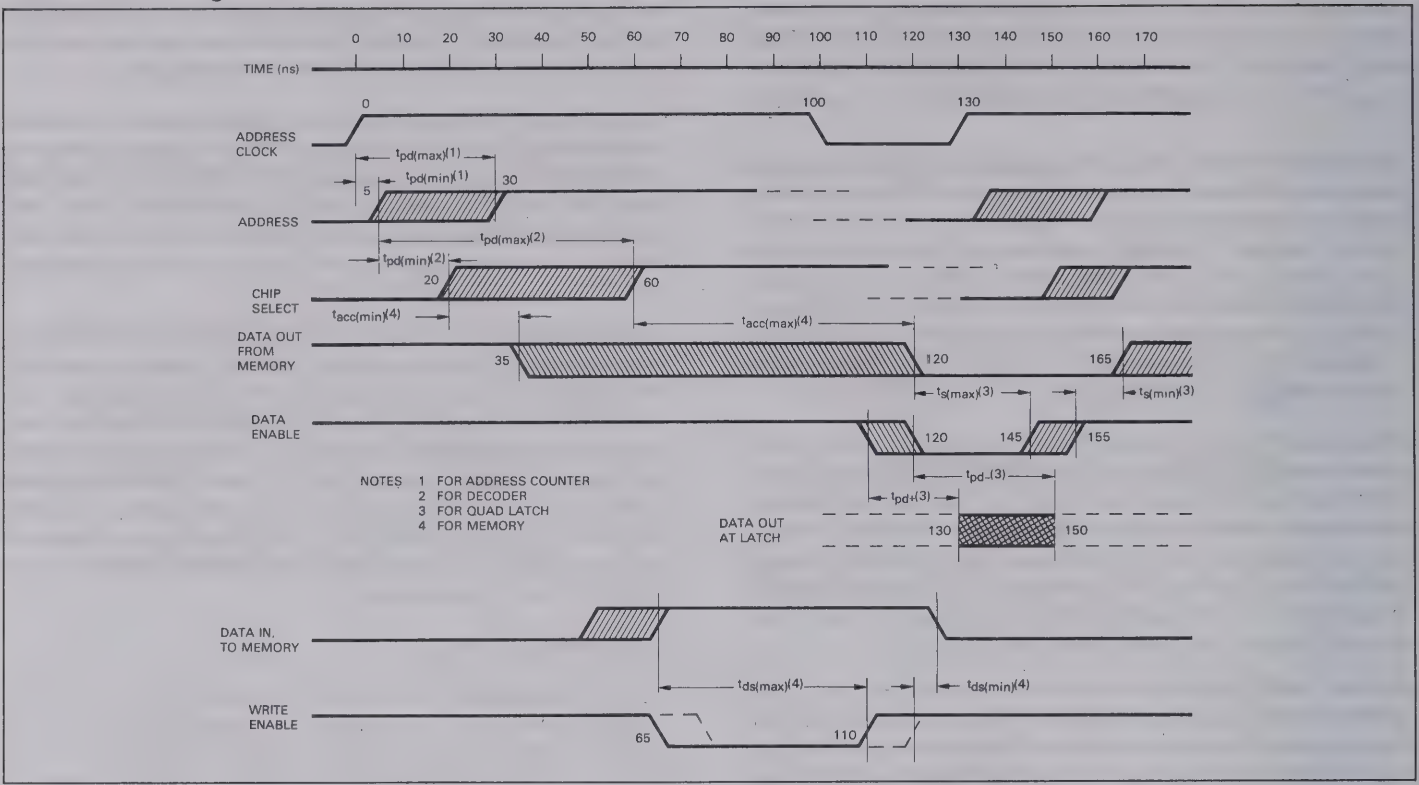
In designing the timing of a memory system within the framework of these eight device delays, it is best to start by showing all the signals relative to a single time axis in a timing diagram. Only one pulse train can be precisely known at all times, and that pulse train must be taken as the reference. All other signals will show some uncertainty relative to this reference. (See Transition Time Uncertainties Chart)

To illustrate the foregoing discussion, consider the task of implementing the scratchpad function in a typical minicomputer requiring temporary storage of 64 4-bit characters. The following block diagram of one possible logic configuration shows how all the different delays can be balanced against one another. Using the approach defined in the Fast Access Timing Chart permits fast operation at the expense of complex timing signal generation. The other chart shows an alternate approach which simplifies timing signal source design but sacrifices speeds.

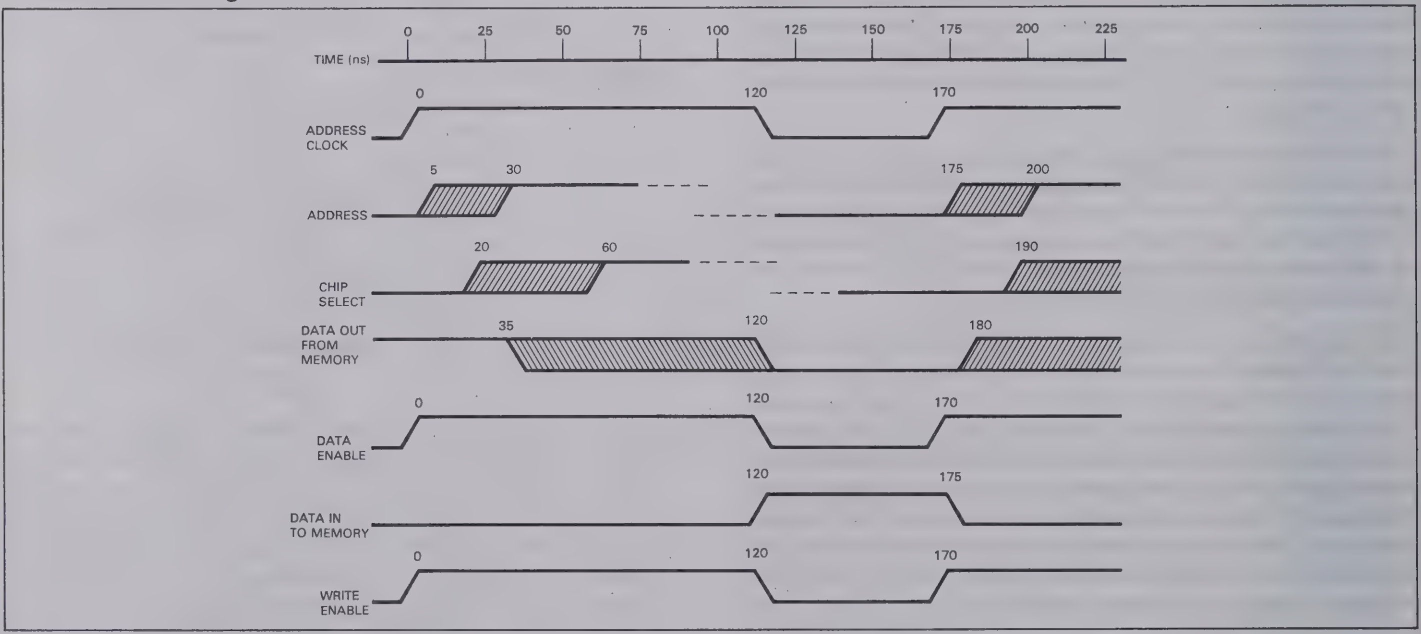
Scratchpad Memory



Fast Access Timing Chart


TIME (ns)
0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170
ADDRESS CLOCK
0
100
130
tpd(max)(1)
tpd(min)(1)
5
30
ADDRESS
tpd(max)(2)
tpd(min)(2)
20
60
CHIP SELECT
tacc(min)(4)
tacc(max)(4)
DATA OUT FROM MEMORY
35
120
165
ts(max)(3)
ts(min)(3)
DATA ENABLE
120
145
155
tpd-(3)
tpd+(3)
NOTES 1 FOR ADDRESS COUNTER
2 FOR DECODER
3 FOR QUAD LATCH
4 FOR MEMORY
DATA OUT AT LATCH
130
150
DATA IN, TO MEMORY
tds(max)(4)
tds(min)(4)
WRITE ENABLE
65
110

Conventional Timing Chart


TIME (ns)
0 25 50 75 100 125 150 175 200 225
ADDRESS CLOCK
0
120
170
ADDRESS
5
30
175
200
CHIP SELECT
20
60
190
DATA OUT FROM MEMORY
35
120
180
DATA ENABLE
0
120
170
DATA IN TO MEMORY
120
175
WRITE ENABLE
0
120
170

INTERFACE

In most applications of bipolar memories, the devices are combined with other TTL or ECL logic elements into a subsystem such as a CPU buffer controller or other function. The memory device interface is at standard logic levels, and the additional hardware required is usually limited to pull up resistors at the outputs of most TTL memories, and load resistors or termination resistors for the ECL memories.

In some cases, the application may require location of the memory several feet or more away from the other functions in the subsystem. The resulting requirement for data transmission necessitates additional hardware. Data transmission is discussed in detail in Communication Line Interface and should prove helpful to the designer. In addition, the reader should refer to the Computer/Interface Section of the Fairchild Linear Integrated Circuits Catalog for information on line drivers, line receivers, display decoders/drivers, EIA/MIL interface circuits, and other products useful in interface design.

RANDOM ACCESS READ/WRITE MEMORIES

SCRATCHPAD/FILE APPLICATIONS

A scratchpad, often called a file, is a small memory integrated into the arithmetic section of a CPU and used for temporary storage of current data. Speed is of the essence with cost relatively insignificant. Early files consisted of flip-flop registers, later followed by 4 to 16 word RAM or MSI flip-flop arrays. A number of existing designs are based on the 93403, 16 x 4 TTL RAM. Currently, files of 256 words or more are in development. The 93410A and 95410 are good candidates for new design applications.

MICRO-CONTROL STORAGE USING READ/WRITE MEMORY

Early in semiconductor memory development, a significant amount of attention was devoted to read only memories for micro-control storage. In many cases, difficulties were encountered in developing firmware for new machines. These difficulties involved turnaround time of weeks and months in making firmware changes, with costs ranging from tens to thousands of dollars per change. One solution to these problems is to use RAMs for micro-control storage. Firmware may then be changed almost instantaneously, thus greatly accelerating the development program and eliminating cost and downtime for pattern changes. If desired, conversion from RAM to ROM can be made at the preproduction phase. Availability of 1024 bit bipolar RAMs such as the 93415 and 95415 has prompted numerous designers to consider this approach.

BUFFER MEMORIES

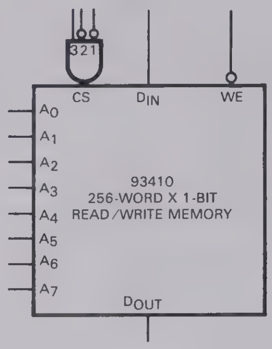
Buffer memories are small to medium memories inserted between I/O interfaces and CPU, between main memory and CPU, or at other locations where fast intermediate storage is required. The availability of 256 and 1024-bit RAM devices has resulted in many bipolar buffer memory designs.

MAIN MEMORIES

Main memories vary from 4K to 16K bits in minicomputers up to 256K or more words in large mainframes. Before the availability of bipolar 1024 RAMs, system designers were limited to low cost core with 1 to 2 μs access, expensive core with 400 ns to 1 μs access, or MOS with >200 ns access. Some n-channel MOS products promise faster access time at low cost. Present bipolar RAM technology allows implementing large main memories with 70 to 100 ns worst case maximum access times for the subsystem. A read-modify-write cycle of less than 150 ns is possible using the 93410 RAM and 93XX/93SXX TTL logic.

93410/93410A RANDOM ACCESS MEMORY

DESCRIPTION AND OPERATION

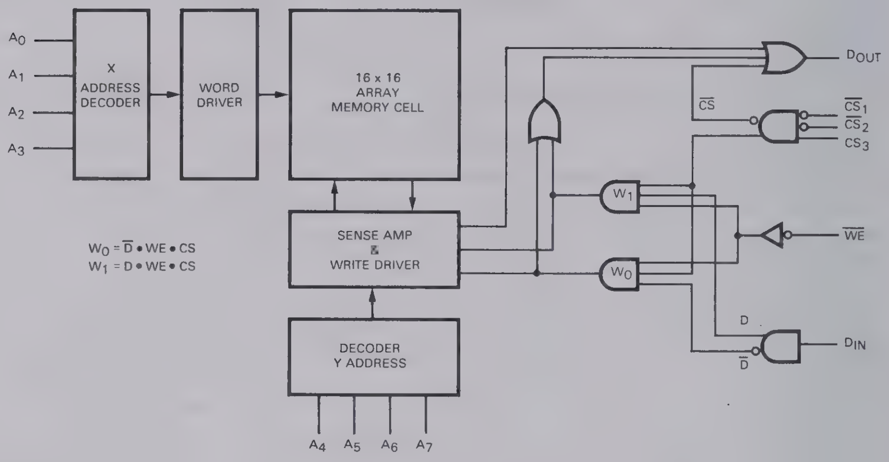


3 2 1
CS
D_{IN}
WE
A_0
A_1
A_2
A_3
A_4
A_5
A_6
A_7
93410
256-WORD X 1-BIT
READ/WRITE MEMORY
D_{OUT}

INPUTS					OUTPUT	MODE
$\overline{CS}_1$	$\overline{CS}_2$	CS_3	$\overline{WE}$	D_{IN}	D_{OUT}	
H	X	X	X	X	H	Not Selected
X	H	X	X	X	H	Not Selected
X	X	L	X	X	H	Not Selected
L	L	H	L	L	H	Write Zero
L	L	H	L	H	H	Write One
L	L	H	H	X	D_{OUT}	Read data from addressed location

LEADS	LOADING
$\overline{CS}_1, \overline{CS}_2, CS_3$	0.5 UL
$A_0 - A_7$	0.5 UL
D_{IN}	0.5 UL
D_{OUT}	10 UL
$\overline{WE}$	0.5 UL

1 UL = 40 μA High/1.6 mA Low
10 UL is the output Low drive factor. An external pull up resistor is needed to provide High level drive capability. This output will sink 16 mA max at V_{OUT} = 0.45 V.



A_0
A_1
A_2
A_3
X ADDRESS DECODER
WORD DRIVER
16 x 16 ARRAY MEMORY CELL
SENSE AMP & WRITE DRIVER
DECODER Y ADDRESS
A_4 A_5 A_6 A_7
$W_0 = \overline{D} \bullet WE \bullet CS$
$W_1 = D \bullet WE \bullet CS$
D_{OUT}
$\overline{CS}$
$\overline{CS}_1$
$\overline{CS}_2$
CS_3
W_1
W_0
$\overline{WE}$
D
$\overline{D}$
D_{IN}

The 93410 and 93410A are high-speed 256-bit TTL random access read/write memories with full decoding on the chip. Each memory, organized as 256 words x 1 bit, is designed for scratchpad, buffer and distributed main memory applications. Both devices have three Chip Select inputs to simplify their use in larger memory systems. Address input lead locations are specifically chosen to permit maximum package density and to provide ease of PC board layout. An uncommitted collector output is available to permit OR-ties for easy memory expansion.

The 93410A is a high speed version of the 93410, offering a 35 ns access time. Since the 93410 and 93410A logic functions are the same, the term 93410 used in the following discussion applies to both types.

As shown in the logic diagram, word selection is achieved with the 8-bit address input, $A_0 - A_7$. Three Chip Select inputs are provided — two active Low ($\overline{CS}_1$ and $\overline{CS}_2$) and one active High (CS_3) — for maximum logic flexibility. This permits memory array expansion up to 2048 words without additional external decoders. For larger memories, the fast Chip Select access time permits the decoding of Chip Select from the address without increasing address access time. The read and write operations are controlled by the state of the active Low Write Enable $\overline{WE}$. With $\overline{WE}$ held Low and the chip selected, the data at D_{IN} is written into the addressed location. To read, $\overline{WE}$ is held High and the chip selected. Data in the specified location is presented at D_{OUT} and is not inverted.

In many applications such as memory expansion, the outputs of many 93410s can be tied together. In other applications the wired-OR is not used. In either case, an external pull up resistor R_L must be used to provide a High at the output when it is off. Any value of R_L within the range specified below may be used.

$$\frac{5.25}{16 - FO\,(1.6)} < R_L < \frac{2.25}{n\,(0.05) + FO\,(0.04)}$$

Where: R_L is in kΩ

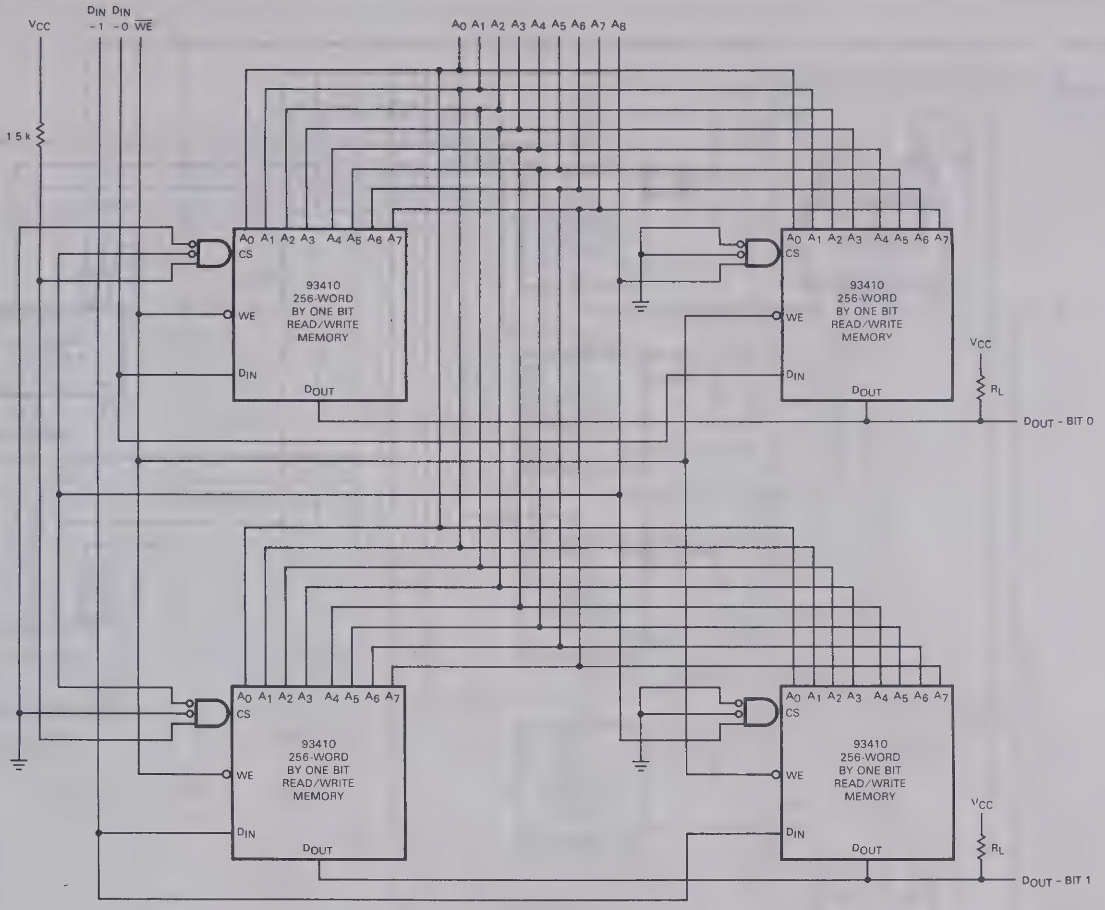
n = number of wired-OR outputs tied together

FO = number of TTL Unit Loads (UL) driven

The minimum value of R_L is limited by output current sinking ability. Maximum R_L is determined by the output and input leakage current which must be supplied to hold the output at the required output high voltage V_{OH}.

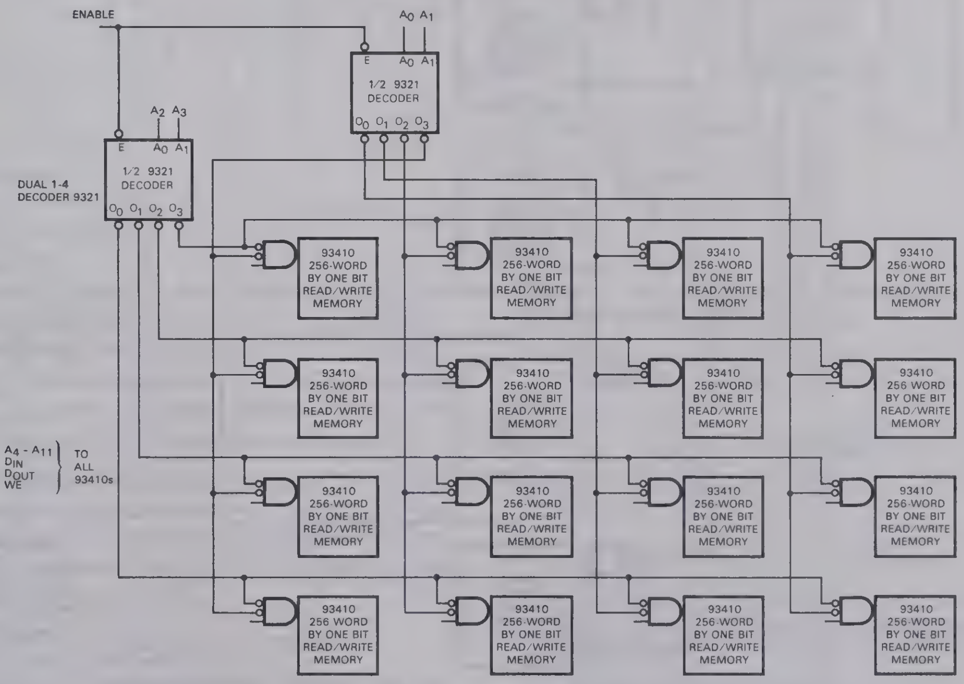
93410/93410A RANDOM ACCESS MEMORY APPLICATIONS

WORD EXPANSION



VCC
DIN -1
DIN -0
WE
A0 A1 A2 A3 A4 A5 A6 A7 A8
1.5 k
A0 A1 A2 A3 A4 A5 A6 A7
CS
WE
DIN
DOUT
93410
256-WORD
BY ONE BIT
READ/WRITE
MEMORY
VCC
RL
DOUT - BIT 0
DOUT - BIT 1

512 - WORD BY 2-BIT ARRAY

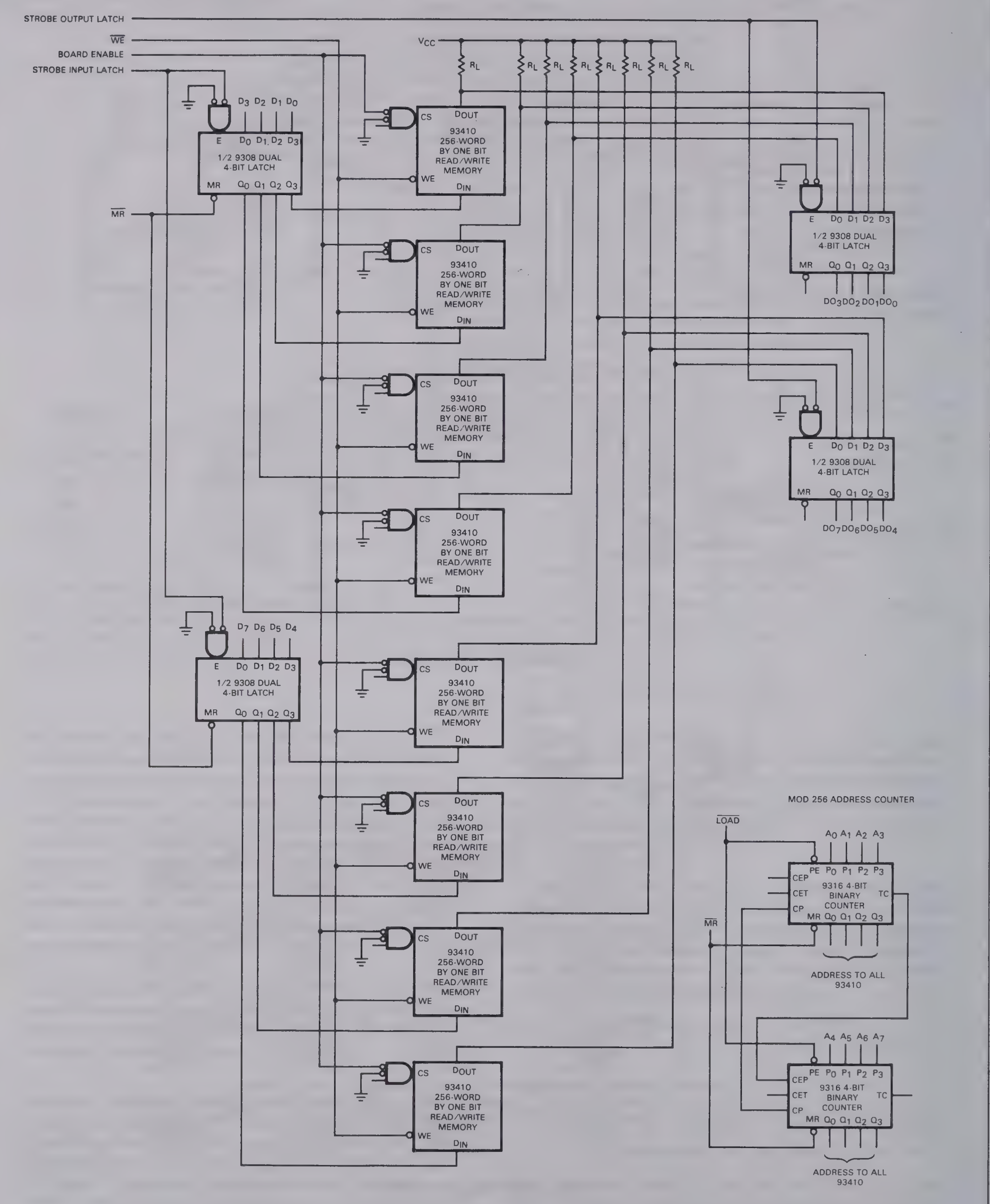


ENABLE
A0 A1
E
A0 A1
1/2 9321
DECODER
O0 O1 O2 O3
A2 A3
DUAL 1-4
DECODER 9321
93410
256-WORD
BY ONE BIT
READ/WRITE
MEMORY
A4 - A11
DIN
DOUT
WE
TO
ALL
93410s

4096 - WORD MEMORY PLANE

The 93410 may be used in memories requiring expansion of both the number of words and number of bits. A 512 x 2 array and the necessary signal interconnects for accomplishing expansion is shown above. The number of words may be expanded to 4096 by using only one 9321 dual 1-of-4 decoder.

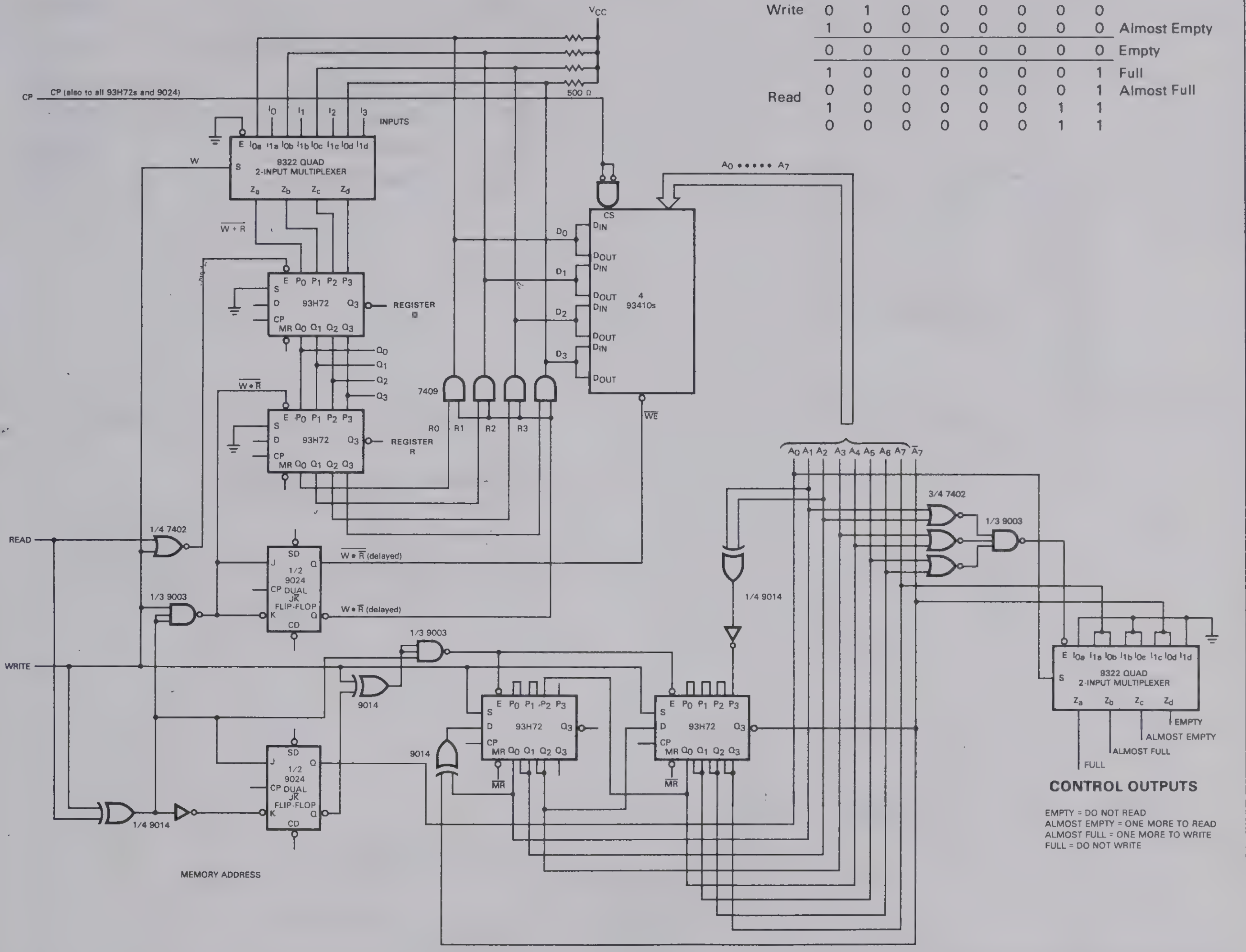
256-WORD BY 8-BIT BUFFER MEMORY SYSTEM


STROBE OUTPUT LATCH
WE
BOARD ENABLE
STROBE INPUT LATCH
MR
VCC
RL
D3 D2 D1 D0
E D0 D1 D2 D3
1/2 9308 DUAL 4-BIT LATCH
MR Q0 Q1 Q2 Q3
CS DOUT
93410 256-WORD BY ONE BIT READ/WRITE MEMORY
WE DIN
D7 D6 D5 D4
DO3 DO2 DO1 DO0
DO7 DO6 DO5 DO4
MOD 256 ADDRESS COUNTER
LOAD
A0 A1 A2 A3
PE P0 P1 P2 P3
CEP
CET
CP
9316 4-BIT BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
ADDRESS TO ALL 93410
A4 A5 A6 A7

A 256-word by 8-bit buffer memory based on the 93410 is shown. Input and output data latches and a modulo 256 address counter may be implemented with MSI devices such as the 9308 quad latch and 9316 binary counter.

LAST IN/FIRST OUT (LIFO) PUSH DOWN STACK MEMORY (254 Words Deep x 4 Bits Wide)

	A_0	A_1	A_2	A_3	A_4	A_5	A_6	A_7	
Write	1	1	0	0	0	0	0	0	
	0	1	0	0	0	0	0	0	
	1	0	0	0	0	0	0	0	Almost Empty
	0	0	0	0	0	0	0	0	Empty
Read	1	0	0	0	0	0	0	1	Full
	0	0	0	0	0	0	0	1	Almost Full
	1	0	0	0	0	0	1	1	
	0	0	0	0	0	0	1	1	


VCC
500 Ω
CP
CP (also to all 93H72s and 9024)
I0
I1
I2
I3
INPUTS
E
S
W
9322 QUAD 2-INPUT MULTIPLEXER
Za Zb Zc Zd
W + R
93H72
REGISTER Q
REGISTER R
Q0
Q1
Q2
Q3
7409
R0 R1 R2 R3
D0
D1
D2
D3
CS
DIN
DOUT
4 93410s
WE
A0 ••••• A7
A0 A1 A2 A3 A4 A5 A6 A7 A7
3/4 7402
1/3 9003
1/4 9014
READ
1/4 7402
1/3 9003
W • R (delayed)
W • R (delayed)
1/2 9024 DUAL JK FLIP-FLOP
SD
CD
WRITE
9014
1/4 9014
MR
MEMORY ADDRESS
9322 QUAD 2-INPUT MULTIPLEXER
EMPTY
ALMOST EMPTY
ALMOST FULL
FULL

CONTROL OUTPUTS

EMPTY = DO NOT READ
ALMOST EMPTY = ONE MORE TO READ
ALMOST FULL = ONE MORE TO WRITE
FULL = DO NOT WRITE

This synchronous memory system accepts data on four parallel inputs ($I_0 - I_3$) and, controlled by two independent inputs (Read and Write), presents the "youngest" word that has not yet been read on the four outputs ($Q_0 - Q_3$). It also provides status information on four outputs: Full, Almost Full, Empty, Almost Empty.

Operation is synchronous and edge triggered on Data as well as Control inputs. It depends on the state of the $I_0 - I_3$, Read and Write inputs, and a setup time ($\approx$30 ns) before the rising edge of the clock that should not exceed 15 MHz at 50% duty cycle.

There are four different modes of operation:

$W \bullet \overline{R}$ = WRITE — I is shifted into Q, the old information in Q is shifted into R, the address counter is incremented, and on the next clock Low period, the content of R is written into the new memory location.

$\overline{W} \bullet R$ = READ — Data in the wired-OR D is shifted into Q, the information in R is maintained, the address counter is decremented. If the previous clock cycle had executed a Write instruction, then D is controlled by the register R. If the previous clock cycle had been one of the other three modes, then D is controlled by the memory.

$W \bullet R$ = READ & WRITE SIMULTANEOUSLY — Input data is shifted into Q; register R and address counter are maintained.

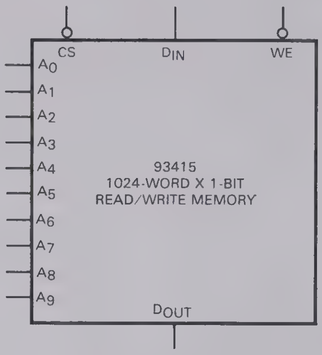
$\overline{W} \bullet \overline{R}$ = DO NOTHING — No change.

The control outputs allow normal computer "handshaking", and also supply a warning signal one operation in advance.

The synchronous up/down address counter is built as a shift register counter. This is both faster and more economical than using 9366 binary counters. The non-binary count sequence is no drawback in this application, and the sacrifice of two of the 256 states is insignificant.

93415 RANDOM ACCESS MEMORY

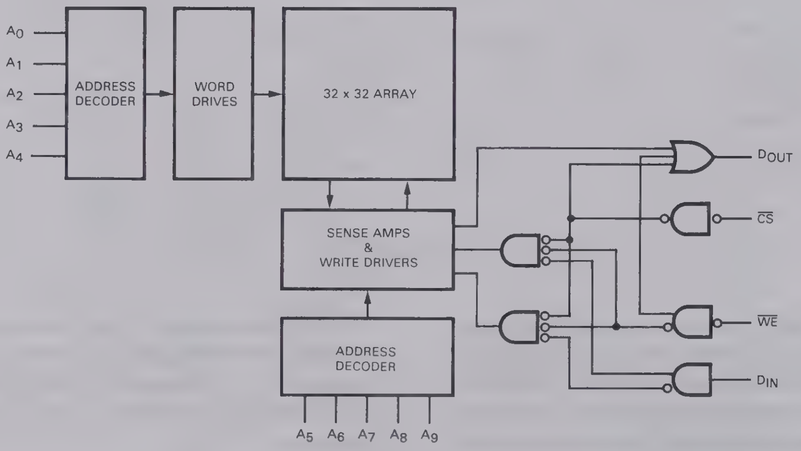
DESCRIPTION AND OPERATION



CS
D_{IN}
WE
A_0
A_1
A_2
A_3
A_4
A_5
A_6
A_7
A_8
A_9
93415
1024-WORD X 1-BIT
READ/WRITE MEMORY
D_{OUT}

LEADS		LOADING
$\overline{CS}$	Chip Select (Active Low)	0.25 UL
$A_0 - A_9$	Address Inputs	0.25 UL
$\overline{WE}$	Write Enable (Active Low)	0.25 UL
D_{IN}	Data Input	0.25 UL
D_{OUT}	Data Output	10 UL

INPUTS			OUTPUT	MODE
$\overline{CS}$	$\overline{WE}$	D_{IN}	Open Collector	
H	X	X	H	Not Selected
L	L	L	H	Write Zero
L	L	H	H	Write One
L	H	X	D_{OUT}	Read



A_0
A_1
A_2
A_3
A_4
ADDRESS DECODER
WORD DRIVES
32 x 32 ARRAY
SENSE AMPS & WRITE DRIVERS
ADDRESS DECODER
A_5
A_6
A_7
A_8
A_9
D_{OUT}
$\overline{CS}$
$\overline{WE}$
D_{IN}

The 93415 is a fully decoded 1024-bit TTL read/write memory organized 1024 words by 1 bit. Bit selection is achieved with a 10-bit address $A_0 - A_9$. One Chip Select input is provided for memory array expansion up to 2048 words by using one external inverter. For larger memories, the fast Chip Select access time permits the decoding of Chip Select $\overline{CS}$ from the Address without affecting system performance. The read and write operations are controlled by the state of the active Low Write Enable $\overline{WE}$. With $\overline{WE}$ held Low and the chip selected, the data at D_{IN} is written into the addressed location. To read, $\overline{WE}$ is held High and the chip selected. Data in the specified location is presented at D_{OUT} and is not inverted. Uncommitted collector outputs are provided to allow maximum flexibility in output connection. In many applications such as memory expansion, the outputs of many 93415s can be tied together. In other applications, the wired-OR is not used. In either case, an external pull up resistor must be used to provide a High at the output when it is off. R_L is calculated using the same formula as shown for the 93410.

The primary advantages of the 93415 over the 93410 are higher density and lower power achieved by making a speed tradeoff. Refer to data sheet parameters to determine specific advantages in each application.

READ ONLY MEMORIES

Read only memories are fixed-contents memories used for random code conversion, for table look-up of certain mathematical functions (multiplication, sine log) and for storing and decoding microinstructions.

BIPOLAR ROMs

Bipolar ROMs offer fast operation, high output drive capability, and operate from a single +5 V supply. Fairchild offers two mask-programmed bipolar read only memories — the 93434, organized 32 words by 8 outputs, and the 93406, organized 256 words by 4 outputs. Both have open collector outputs, chip select inputs (programmable on the 93406) and a through delay of 50 ns max.

MOS ROMs

Fairchild also offers a silicon gate MOS read only memory, the 3524, organized 512 words by 8 outputs. MOS offers greater packing density and lower cost per bit at the expense of an extra supply voltage (–12 V), lower output drive capability and considerably longer access time (650 ns max). The 64 x 5 x 7 and 64 x 5 x 9 silicon gate MOS character generators (3257, 3258, 3260) are specialized ROMs with on-chip counters, used to display 64 ASCII characters on raster scan television displays.

EXPANDING ROMs

The two popular ROM formats, 32 words by 8 outputs, and 256 words by 4 outputs, often have to be expanded into bigger arrays with more addresses and/or more outputs. How to do this efficiently is explained in the applications of the 93434/93406 ROM.

CHANGING THE ROM FORMAT

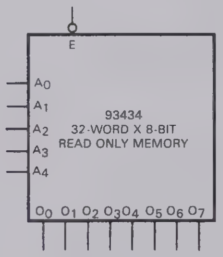
Many applications need more addresses, but fewer outputs, or they need fewer addresses, but more outputs than provided by the available ROMs. Applications are included to show how the format of the 93434/93406 ROM can be modified to meet these requirements.

SELECTIVE POWER DOWN

The power consumption of large ROM arrays can be reduced significantly by applying power, V_{CC}, only to the selected ROMs. However, this technique increases the access time.

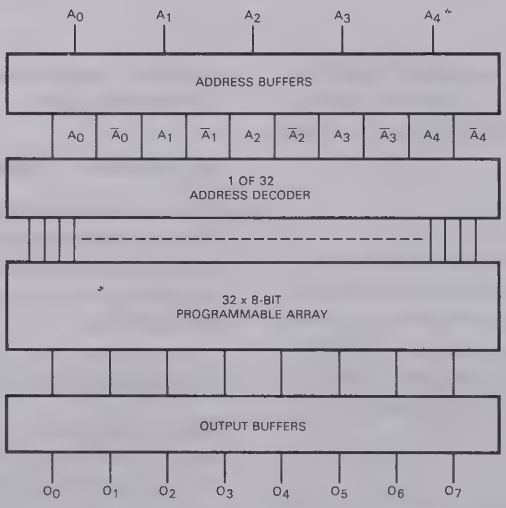
93434 READ ONLY MEMORY

DESCRIPTION


E
A0
A1
A2
A3
A4
93434
32-WORD X 8-BIT
READ ONLY MEMORY
O0 O1 O2 O3 O4 O5 O6 O7

LEADS		LOADING HIGH	LOADING LOW
$A_0 - A_4$	Address Inputs	2 UL	1 UL
$\overline{E}$	Enable (Active Low)	2 UL	1 UL
$O_0 - O_7$	Outputs	O.C.	6.25 UL

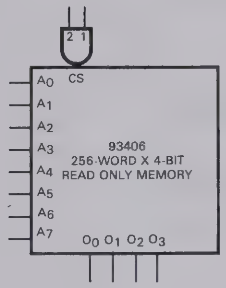
Propagation delay $\overline{E}$ or A_X to O_X = 30 ns (typ), 50 ns (max)


A0
A1
A2
A3
A4
ADDRESS BUFFERS
A0
Ā0
A1
Ā1
A2
Ā2
A3
Ā3
A4
Ā4
1 OF 32
ADDRESS DECODER
32 x 8-BIT
PROGRAMMABLE ARRAY
OUTPUT BUFFERS
O0
O1
O2
O3
O4
O5
O6
O7

The 93434 is a 256-bit bipolar TTL read only memory, organized as 32 8-bit words. The words are selected by five address lines with full decoding included on the chip. The eight outputs are uncommitted collectors which may be wired-OR with other ROMs or other TTL logic devices. When the active Low Enable (Chip Select) input is High, all outputs are High. The contents of the memory are permanently programmed to customer order.

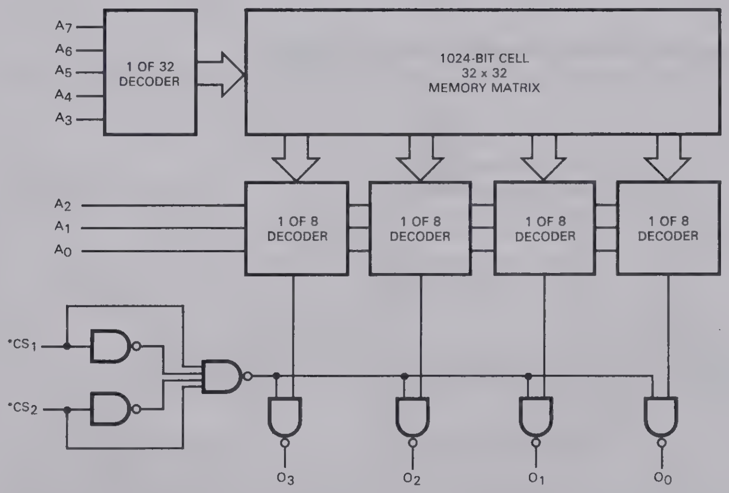
93406 READ ONLY MEMORY

DESCRIPTION


2 1
CS
A0
A1
A2
A3
A4
A5
A6
A7
93406
256-WORD X 4-BIT
READ ONLY MEMORY
O0 O1 O2 O3

LEADS		LOADING HIGH	LOADING LOW
$A_0 - A_7$	Address Inputs	1 UL	0.5 UL
CS_1, CS_2	Chip Select Inputs	1 UL	0.5 UL
$O_0 - O_3$	Outputs	Open Collector	9.3 UL

Propagation delay CS or A_X to O_X = 30 ns (typ), 50 ns (max)

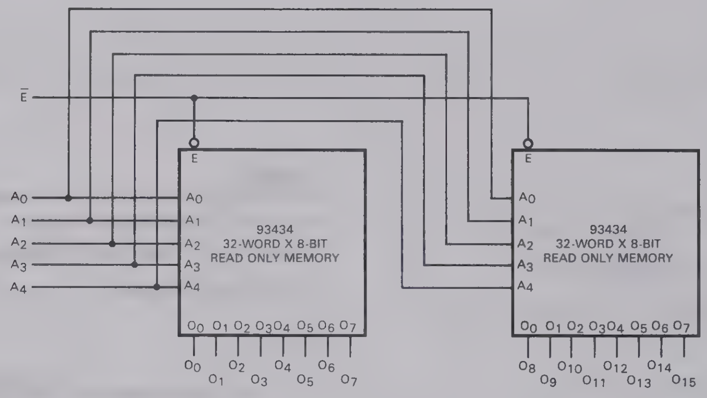

A7
A6
A5
A4
A3
1 OF 32 DECODER
1024-BIT CELL 32 x 32 MEMORY MATRIX
A2
A1
A0
1 OF 8 DECODER
1 OF 8 DECODER
1 OF 8 DECODER
1 OF 8 DECODER
*CS1
*CS2
O3
O2
O1
O0

*Chip Select level is programmable per customer requirements. If not specified, CS_1 and CS_2 are active Low.

The 93406 is organized 256 words by four bits, with eight address inputs and full decoding on the chip. Chip Select leads are programmable to customer selection of any of four patterns thus reducing external decoding requirements. When the Chip Select logic is not True, all outputs are High. Refer to the Fairchild TTL Catalog or 93406 data sheet for details on user generated truth table format.

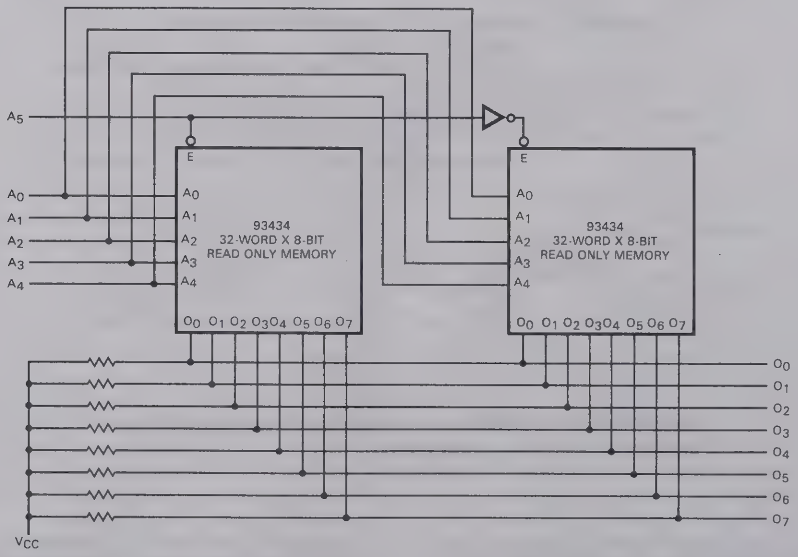
93434/93406 READ ONLY MEMORY APPLICATIONS

EXPANDING ROMs — INCREASING THE NUMBER OF OUTPUTS



Ē
E
A0
A1
A2
A3
A4
93434
32-WORD X 8-BIT
READ ONLY MEMORY
O0 O1 O2 O3 O4 O5 O6 O7
O8 O9 O10 O11 O12 O13 O14 O15

EXPANDING ROMs — INCREASING THE NUMBER OF ADDRESSES



A5
E
A0
A1
A2
A3
A4
93434
32-WORD X 8-BIT
READ ONLY MEMORY
O0 O1 O2 O3 O4 O5 O6 O7
VCC

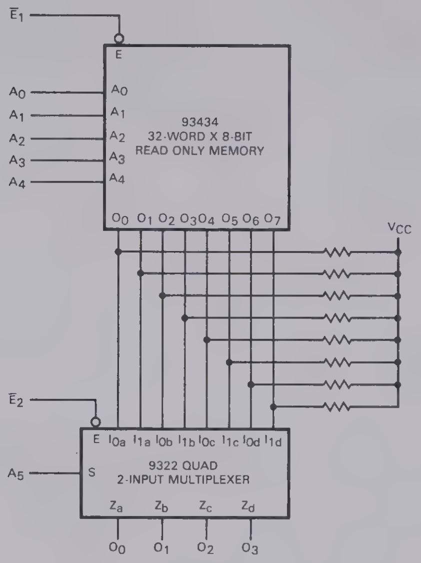
The open-collector outputs are OR-wired and the additional address bit is used to select the individual ROM. This concept can be expanded to more parallel ROMs.

The 93434 then requires a decoder to control the Enable inputs, while the 93406 can decode two address lines with its programmable chip select.

93434/93406 READ ONLY MEMORY APPLICATIONS

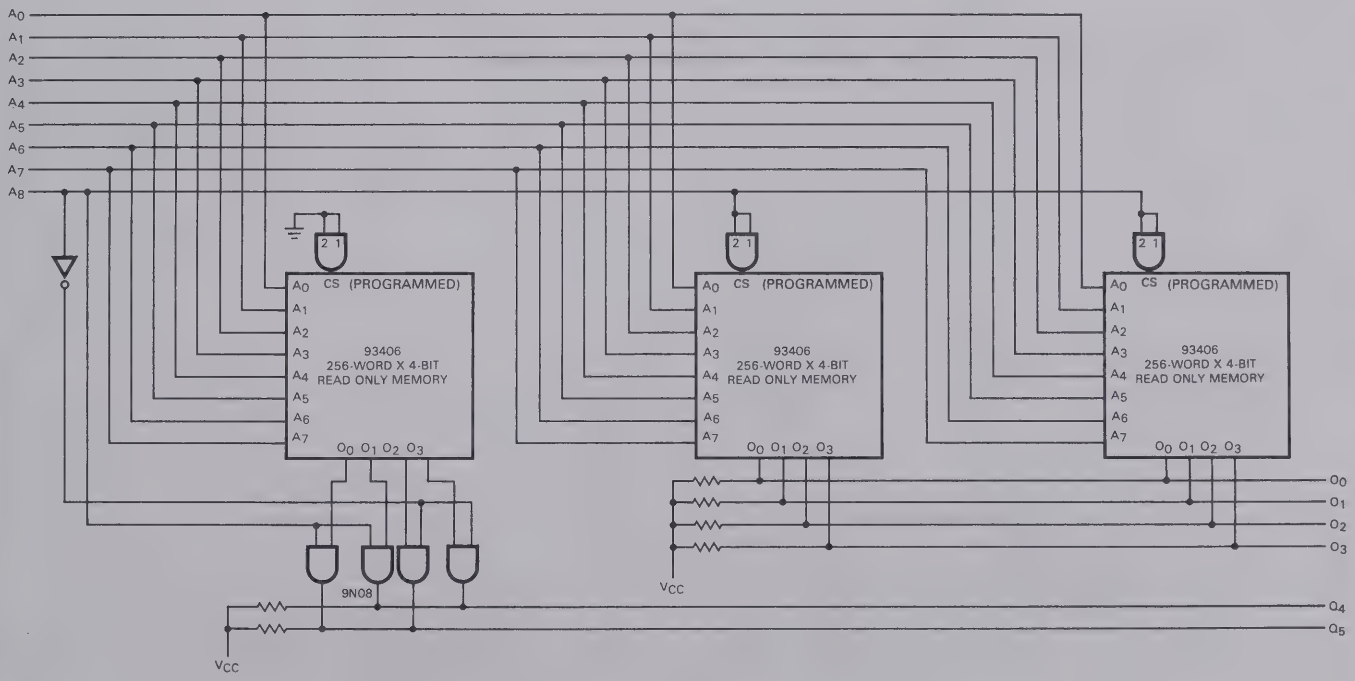
CHANGING THE ROM FORMAT — MORE ADDRESSES, FEWER OUTPUTS

64 WORDS BY 4 OUTPUTS ROM



$\overline{E}_1$
E
A_0
A_1
A_2
A_3
A_4
93434
32-WORD X 8-BIT
READ ONLY MEMORY
O_0 O_1 O_2 O_3 O_4 O_5 O_6 O_7
V_{CC}
$\overline{E}_2$
E I_{0a} I_{1a} I_{0b} I_{1b} I_{0c} I_{1c} I_{0d} I_{1d}
A_5
S
9322 QUAD
2-INPUT MULTIPLEXER
Z_a Z_b Z_c Z_d
O_0 O_1 O_2 O_3

512 WORDS BY 6 OUTPUTS ROM

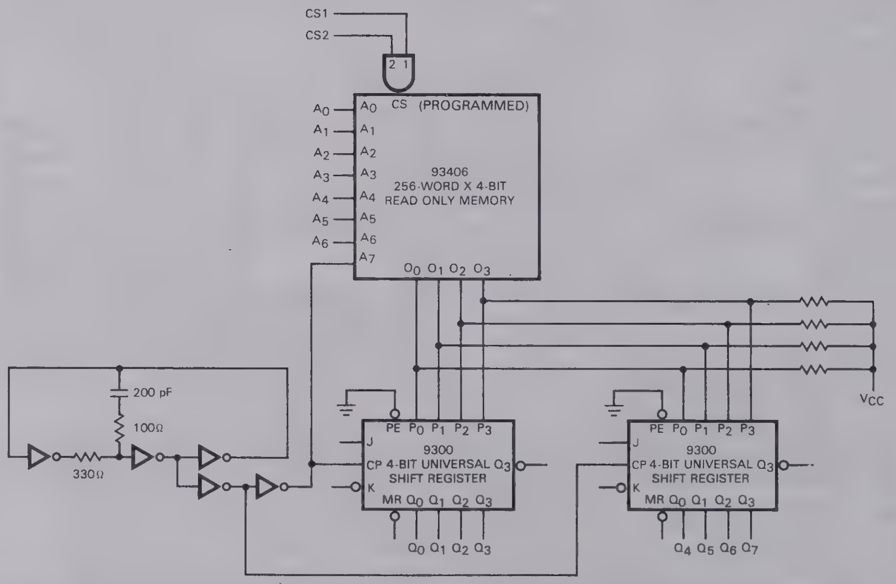


A_0
A_1
A_2
A_3
A_4
A_5
A_6
A_7
A_8
2 1
A_0 CS (PROGRAMMED)
A_1
A_2
A_3
A_4
A_5
A_6
A_7
93406
256-WORD X 4-BIT
READ ONLY MEMORY
O_0 O_1 O_2 O_3
9N08
V_{CC}
O_0
O_1
O_2
O_3
Q_4
Q_5

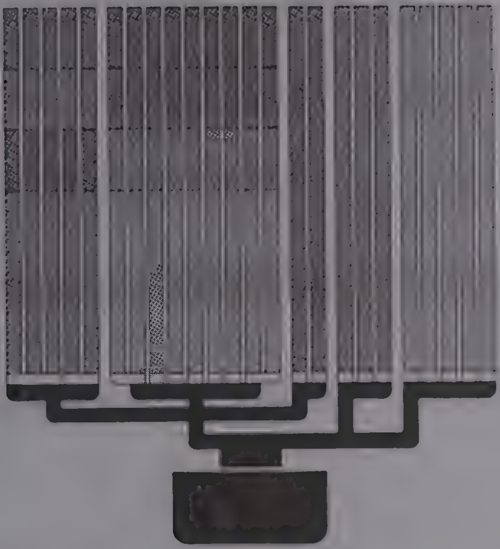
The 93434 32 x 8 ROM can be converted into a 64 x 4 ROM by using a quad 2-input multiplexer (9322). A High level on the E_1 input forces all outputs High, but a High level on the E_2 input overrides and forces all outputs Low.

Three 256 x 4 ROMs can be interconnected to form a 512 x 6 ROM without wasting any bits. This concept can be modified to efficiently provide many other unconventional configurations.

CHANGING THE ROM FORMAT — MORE OUTPUTS, FEWER ADDRESSES


CS1
CS2
2 1
A0
A1
A2
A3
A4
A5
A6
A0
A1
A2
A3
A4
A5
A6
A7
CS (PROGRAMMED)
93406
256-WORD X 4-BIT
READ ONLY MEMORY
O0 O1 O2 O3
200 pF
100Ω
330Ω
VCC
PE P0 P1 P2 P3
J
9300
CP 4-BIT UNIVERSAL Q3
SHIFT REGISTER
K
MR Q0 Q1 Q2 Q3
Q0 Q1 Q2 Q3
PE P0 P1 P2 P3
J
9300
CP 4-BIT UNIVERSAL Q3
SHIFT REGISTER
K
MR Q0 Q1 Q2 Q3
Q4 Q5 Q6 Q7

A 256 x 4 ROM can be converted into a 128 x 8 ROM, but this change in format requires two ROM access operations for each code conversion. A 5 MHz oscillator is used to modify the address, and two 4-bit universal registers (9300) assemble the output information. This increases the apparent access time to 200 ns.



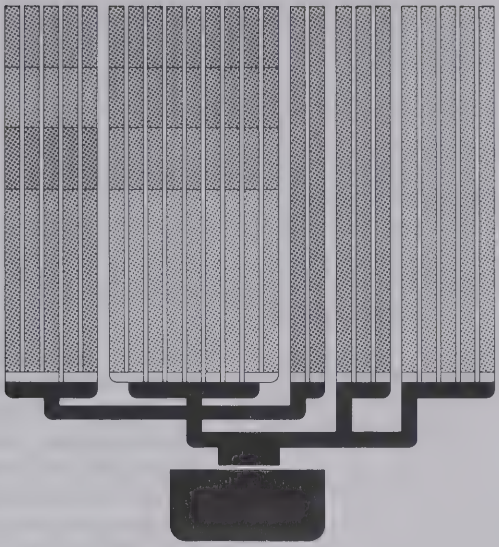
Registers

- Quad D Flip-Flop and Dual 2-Bit Register
- Left/Right Shift Register
- Serial-to-Parallel Converter
- Dual Input Shift Register
- Recirculation
- Edge Detector with Storage
- Parallel-to-Serial Converter
- Variable Length Shift Register
- Serial Memories
- Multiplexed 7-Segment Count and Display
- Modulo 16 Counter
- Pseudo-Random Sequence Generators
- Arithmetic Logic Unit
- First In/First Out Memory

REGISTER SELECTION GUIDE

S — Synchronous
A — Asynchronous
P — Preset (Asynchronous ones transfer only)

Function	Device	# of Bits	Serial Entry	Parallel Entry # Of Bits	# Of Outputs	# Of Clocks	Clock Edge	Max. Clock Freq. MHz (typ)	Clock To Output Delay,ns (typ)	Power Dissipation mW (typ)	Leads
Parallel In Parallel Out Shift Right	9300,74195	4	$J,\overline{K}$	4 S	4 + 1	1	↑	38	16	300	16
	93H00	4	$J,\overline{K}$	4 S	4 + 1	1	↑	55	12	325	16
	93L00	4	$J,\overline{K}$	4 S	4 + 1	1	↑	17	28	75	16
	93S00,74S195	4	$J,\overline{K}$	4 S	4 + 1	1	↑	105	10	350	16
	93H72	4	D	4 S	4 + 1	1	↑	60	12	475	16
	93178,74178	4	D	4 S	4	1	↓	39	23	230	14
	93179,74179	4	D	4 S	4 + 1	1	↓	39	23	230	16
	93199,74199	8	$J,\overline{K}$	8 S	8	2 (OR)	↑	35	20	360	24
	9395,7495	4	D	4 S	4	2 (S,P)	↓	36	20	195	14
	9396,7496	5	D	5 P	5	1	↑	10	25	240	16
Parallel In Parallel Out BiDirectional	93194,74194	4	D_R,D_L	4 S	4	1	↑	36	17	195	16
	93S194,74S194	4	D_R,D_L	4 S	4	1	↑	105	10	425	16
	93198,74198	8	D_R,D_L	8 S	8	1	↑	35	20	360	24
Serial In Parallel Out	93164,74164	8	D_A OR D_B		8	1	↑	36	19	185	14
Parallel In Serial Out	9394,7494	4	D	(4 + 4) P	1	1	↑	10	25	175	16
	93165,74165	8	D	8 A	1 + 1	2 (OR)	↑	26	19	210	16
	93166,74166	8	D	8 S	1	2 (OR)	↑	35	20	360	16
Serial In	9328	16	2X (2-Input MUX)		2 + 2	3 (OR)	↑	30	17	300	16
Serial Out	93L28	16	2X (2-Input MUX)		2 + 2	3 (OR)	↑	15	42	80	16
	9391	8	D_A AND D_B		1 + 1	1	↑	18	25	175	14
Multiport Register	9338	8	D		2	1	↑	25	23	425	16
	93S39	8	D		3	1	↑				16



REGISTERS

INTRODUCTION

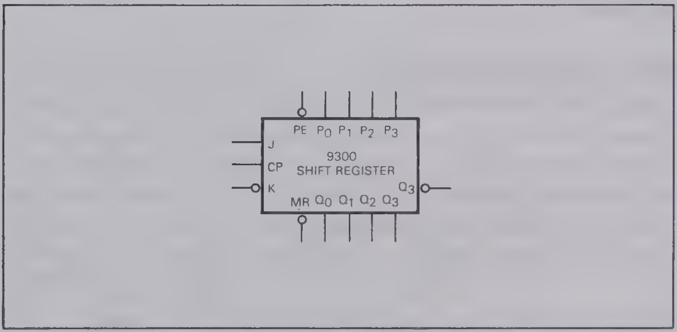
Registers are data storage devices that are more sophisticated than latches. They use edge triggered flip-flops (see the SSI Chapter for detailed descriptions) and are therefore non-transparent, i.e., the outputs change as a result of a clock edge according to input signals that were present before this clock edge. Outputs can therefore be fed back to inputs without incurring any race conditions. (The asynchronous data inputs of the 9394 and 9396 registers do not follow this rule and must be used more carefully.)

There are 13 registers in the Fairchild TTL/MSI family, and they can be classified into four groups.

- 4 and 5-bit general purpose shift registers for general storage, shifting, counting and code conversion.
- 8-bit shift registers for parallel-to-serial and serial-to-parallel conversion.
- 8 and dual 8-bit (16-bit) shift registers for serial data storage.
- An 8-bit multiple port register for multiple access storage.

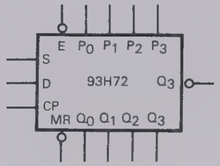
The selection guide summarizes the features of all of these registers. The largest number of registers are in the 4-bit general purpose category.

9300, 93L00, 93H00, 93S00 4-Bit Shift Registers



The 9300 is a universal 4-bit shift register with serial and parallel synchronous operating modes. It has parallel inputs and outputs on each stage, a complement output on the last stage, an active Low overriding Master Reset and J$\overline{K}$ input configuration for increased functional capability in the serial mode. The 93L00 is the low power version (10 MHz at 75 mW) of the standard 9300 (38 MHz at 300 mW). The 93H00 is the high speed version (55 MHz at 325 mW). The 93S00 is a super high speed Schottky TTL version (100 MHz). The 93L00, 93H00 and 93S00 are all logic equivalents to the 9300.

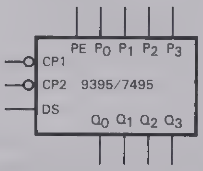
93H72 4-Bit Shift Register With Enable



The 93H72 is a 60 MHz shift register similar to the 9300 with three synchronous modes of operation: Shift, Parallel Load and Hold (do nothing). (The 9300 cannot "do nothing" without sacrificing its parallel load capability, but it has a more flexible serial input structure, J$\overline{K}$.) The device has parallel inputs and outputs on each stage, an overriding active Low Master Reset and a single line D-type input for serial data entry. A High on the Enable input overrides shifting and parallel load and holds information in the register.

The synchronous parallel load capability of the 9300 (93L00 and 93H00) and 93H72 makes these devices flexible and easy to use for a wide variety of applications. The 9300 or variations (93L00, 93H00) are preferred for applications where the serial J$\overline{K}$ input flexibility is needed or where power and speed requirements dictate its use. The 93H72 should be used when information must be parallel loaded, shifted and held.

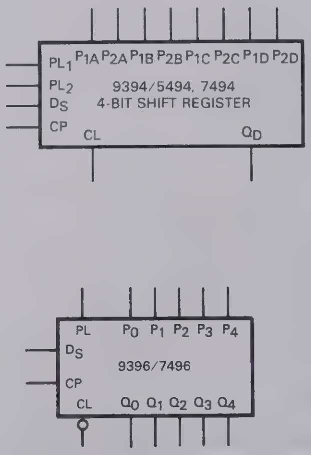
9395 4-Bit Shift Register



The 9395 is similar to the 9300 or 93H72 in that parallel and serial data operation is synchronous. However, unlike the 9300, the 9395 has two active Low clocks that provide independent clocking for serial and parallel data entry. Also, the mode polarities are reversed from those of the 9300, with parallel data entry occurring when the mode line (PE) is High on the 9395 (Low on the 9300).

Since the 9395 shifts and loads synchronously it is as flexible in this respect as the 9300, 93L00 and 93H72. However, mode change restrictions, the lack of a Master Reset, J$\overline{K}$ inputs and non-standard clock polarities (outputs change on a High-to-Low transition) must be acceptable. In some cases the independent clock control for serial and parallel operation, and the high output drive capability (10 unit loads) can be used to advantage.

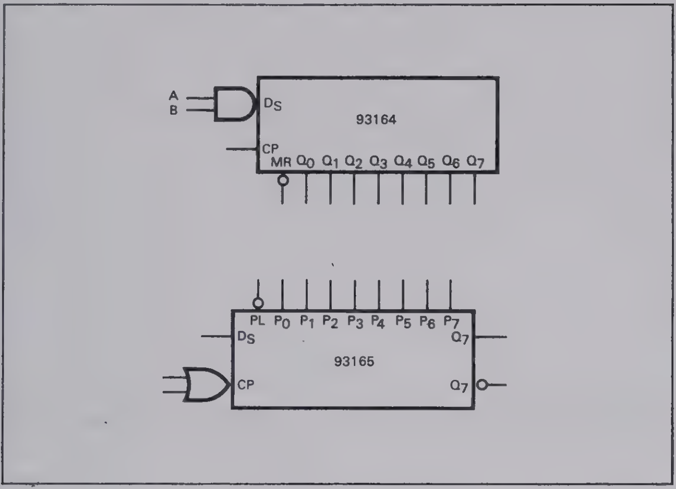
9394 And 9396 4 And 5-Bit Shift Registers



The 9394 and 9396 are 4 and 5-bit shift registers with synchronous shifting and non-synchronous ones transfer parallel data entry. Both registers are resettable and both have positive edge clocks, D-type serial inputs and output drive capabilities of 10 unit loads. When the parallel load control is activated, ones on the parallel inputs force the register bits to one, but the parallel inputs cannot force the register bits to zero. Two parallel load controls and two sets of inputs are provided on the 9394 to facilitate parallel ones transfer from two data sources. But, in order to accommodate the extra set of inputs, only the output of the last register bit is brought out. Outputs are available from each stage of the 9396.

The ones transfer type of parallel data entry lends itself to asynchronous ones-catching applications but is cumbersome for data entry since the registers must be reset (through the asynchronous reset or through the serial entry of zeros) before loading parallel data (ones). The 9396 is the only 5-bit shift register available and can save packages in certain applications.

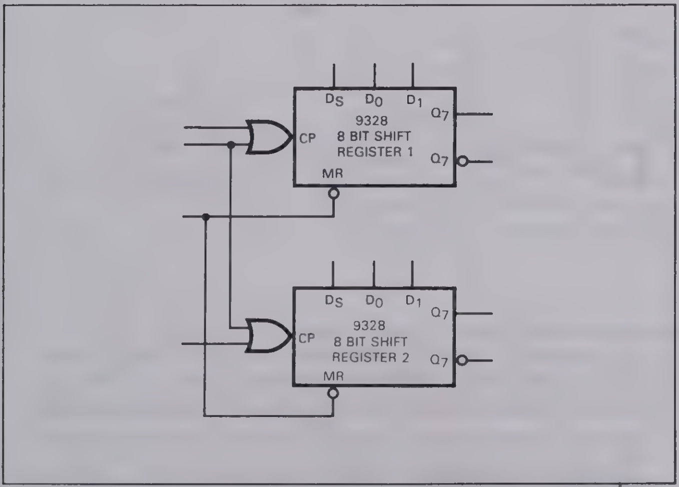
93164 Serial-To-Parallel And 93165 Parallel-To-Serial Converters



The 93164 and 93165 devices are 8-bit shift registers designed for parallel/serial conversions. The 93165 is a parallel-to-serial converter or 8-bit shift register with non-synchronous parallel data inputs, synchronous serial operation, negation and assertion outputs from the last stage and positive edge ORed clock inputs. In addition to the basic parallel-to-serial function, this register can be used as a serial storage register where a special code must be inserted in parallel. The 2-input OR clock can be used to combine two independent clock sources, or one input can act as a clock enable (active Low).

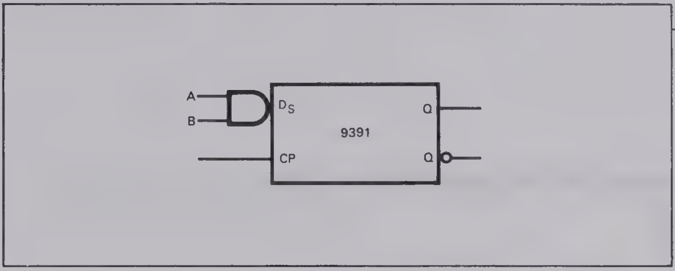
The 93164 is a serial-to-parallel converter or 8-bit shift register with outputs from each stage, AND-type serial data input, positive edge clock and active Low overriding Master Reset. The 93164 can be used in many applications to replace two 4-bit shift registers if parallel entry is not necessary.

9328 And 93L28 Serial Storage Registers



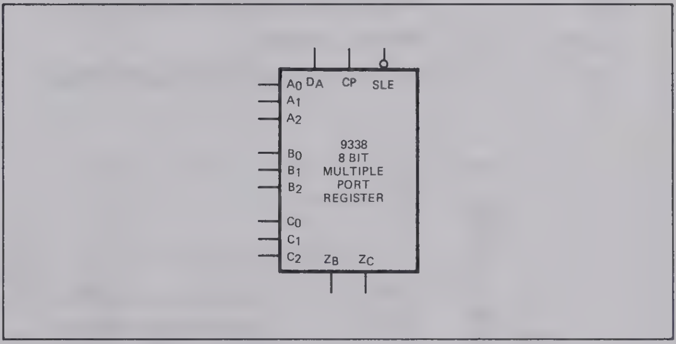
These devices are long registers without parallel outputs suited for serial data storage. The 9328 is a dual 8-bit shift register and the 93L28 provides the identical function in a low power version. Each 8-bit shift register has a True and Complement output available on the last stage, a 2-input multiplexer with data control on the input for data source selection and a 2-input OR clock input. Both registers have a common clock obtained by internally tying one input of each clock OR gate together, and an overriding asynchronous Master Reset.

9391 8-Bit Shift Register



The 9391 is an 8-bit shift register with D-type AND serial input, complementary outputs and a positive edge clock. It is an alternative to the 9328 when only eight bits of serial storage are required and when non-standard power leads and the lack of a Master Reset can be tolerated.

9338 8-Bit Multiple Port Memory



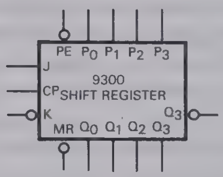
The 9338 is an 8-bit multiple port register. Data can be written into any one of the eight bits and read from any two of the eight bits independently. The device features master/slave operation, so that data is written when the clock is Low and read when the clock is High. The device is designed for use as a 1-bit slice of eight or more working registers/accumulators, or in scratch pad memory applications in a three address processing unit.

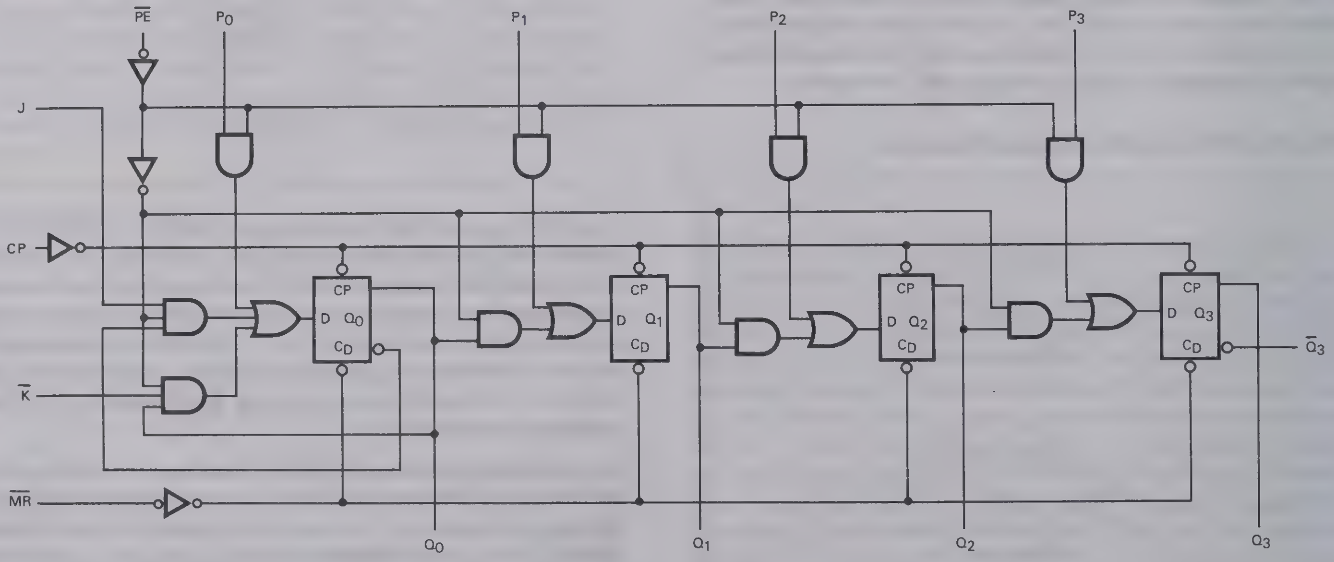
Three address fields are provided: one write field ($A_0A_1A_2$) for data entry and two read fields for data readout ($B_0B_1B_2$, $C_0C_1C_2$). Data on the input is routed to one of eight latches by the write address code A_0, A_1, A_2. Data from any one of these latches can be read out through a slave latch on the Z_B output by applying the desired read address to B_0, B_1 and B_2. Independently, data also can be read out through a slave latch on Z_C from the latch addressed by C_0, C_1 and C_2. Because of the master/slave operation, the write address can be the same as one of the read addresses without incurring a race condition. The Slave Enable control ($\overline{SLE}$) overrides the clock to the slaves. If $\overline{SLE}$ is Low, the slaves are enabled all the time.

9300 SHIFT REGISTER

DESCRIPTION AND OPERATION

LEADS		LOADING
$\overline{PE}$	Parallel Enable (Active Low) Input	2.3 UL
P_0, P_1, P_2, P_3	Parallel Inputs	1 UL
J	First Stage J (Active High) Input	1 UL
$\overline{K}$	First Stage K (Active Low) Input	1 UL
CP	Clock (Active High Going Edge) Input	2 UL
$\overline{MR}$	Master Reset (Active Low) Input	1 UL
Q_0, Q_1, Q_2, Q_3	Parallel Outputs	6 UL
$\overline{Q}_3$	Complementary Last Stage Output	8 UL


PE P0 P1 P2 P3
J
9300
CP SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3


PE
P0
P1
P2
P3
J
CP
K
MR
CP
D Q0
CD
CP
D Q1
CD
CP
D Q2
CD
CP
D Q3
CD
Q3
Q0
Q1
Q2
Q3

The 9300 4-bit Shift Register is a general purpose, sequential MSI unit. Operation of the 9300 Shift Register is indicated by the logic diagram shown here. The device has four clocked master/slave flip-flops with D inputs. The D input of each stage can be switched between two logical sources by the Parallel Enable ($\overline{PE}$) input. When the $\overline{PE}$ input is Low, the D inputs of the four stages (Q_0, Q_1, Q_2 and Q_3) are connected to the parallel inputs (P_0, P_1, P_2 and P_3). These inputs determine the next condition of the shift register synchronously with the clock. With the $\overline{PE}$ input Low, the element appears as four common clocked D flip-flops.

9300 SHIFT REGISTER

MODE SELECTION/DATA ENTRY

$\overline{MR}$	$\overline{PE}$	J	$\overline{K}$	Q_0	Q_1	Q_2	Q_3	
L	X	X	X	L	L	L	L	Asynchronous Reset
H	*h*	*h*	*h*	H	q_0	q_1	q_2	Shift, set first stage
H	*h*	*l*	*l*	L	q_0	q_1	q_2	Shift, reset first stage
H	*h*	*h*	*l*	$\overline{q_0}$	q_0	q_1	q_2	Shift, toggle first stage
H	*h*	*l*	*h*	q_0	q_0	q_1	q_2	Shift, retain first stage
H	*l*	X	X	p_0	p_1	p_2	p_3	Parallel load

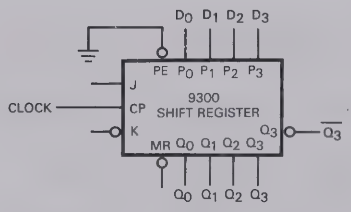
Lower case letters describe the state of the input (or output) one set-up time prior to the Low-to-High clock transition.

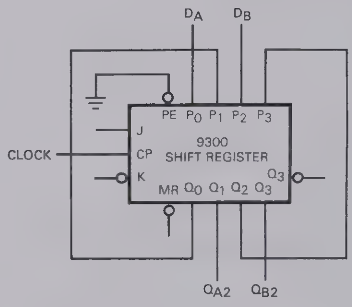
When the $\overline{PE}$ input is High, the D inputs of stages Q_1, Q_2 and Q_3 are connected to the outputs of Q_0, Q_1, Q_2 respectively, forming a 4-bit shift register. The D input to stage Q_0 with $\overline{PE}$ High is obtained from the J and $\overline{K}$ inputs via gating elements to produce the action of the first stage as shown in the truth table. As a result, the shift register performs a 1-bit shift for each clock input. In both the serial and parallel modes the outputs change state after the Low-to-High clock transition. All stages are set to zero when the Master Reset ($\overline{MR}$) input is Low, regardless of any other inputs.

Except in response to the Master Reset input, the outputs of the 9300 shift register change only on Low-to-High transition of the clock input signal. As a result of this synchronous operation, much less external logic is required in most applications. The logic operation performed is determined by the logic levels on the J, $\overline{K}$, P or $\overline{PE}$ inputs and set up time before the rising edge of the clock (non ones-catching). There is no other restriction on the activity of those inputs.

9300 SHIFT REGISTER APPLICATIONS

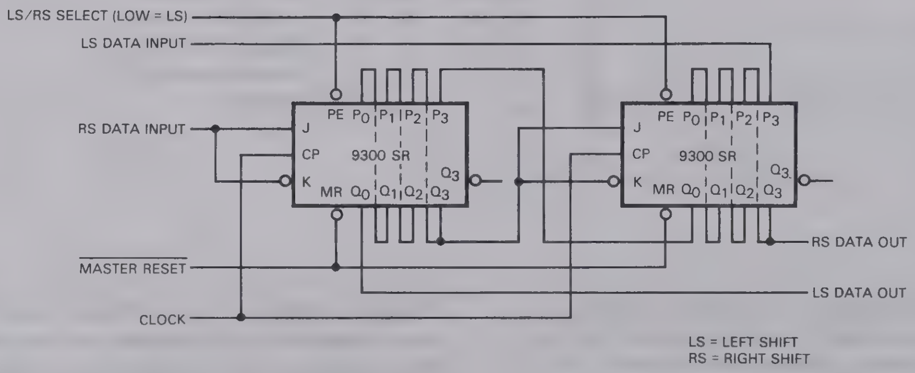
QUAD D FLIP-FLOP OR DUAL 2-BIT REGISTER


D0 D1 D2 D3
PE P0 P1 P2 P3
J
CLOCK
CP
9300
SHIFT REGISTER
K
Q3
Q̄3
MR Q0 Q1 Q2 Q3
Q0 Q1 Q2 Q3


DA DB
PE P0 P1 P2 P3
J
CLOCK
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
QA2 QB2

As mentioned earlier, when the 9300 is operated in the parallel mode it appears as four common-clocked D flip-flops. These four flip-flops can be externally interconnected to form other combinations as in the dual 2-bit configuration above.

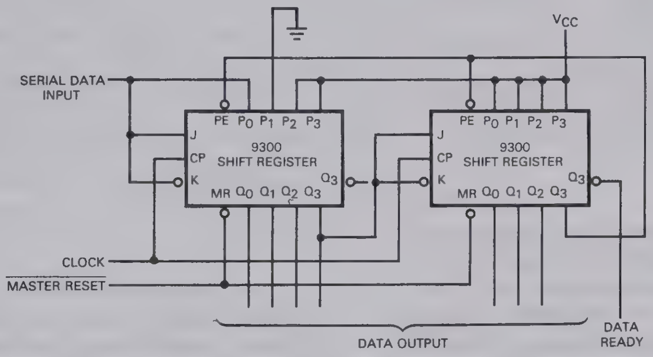
LEFT/RIGHT SHIFT REGISTER


LS/RS SELECT (LOW = LS)
LS DATA INPUT
RS DATA INPUT
MASTER RESET
CLOCK
PE P0 P1 P2 P3
J
CP
9300 SR
K
Q3
MR Q0 Q1 Q2 Q3
PE P0 P1 P2 P3
J
CP
9300 SR
K
Q3
MR Q0 Q1 Q2 Q3
RS DATA OUT
LS DATA OUT
LS = LEFT SHIFT
RS = RIGHT SHIFT

The synchronous parallel inputs of the 9300 can be used to produce a register that shifts left or right on each clock. As shown above, each 9300 has the Q_1, Q_2 and Q_3 outputs connected to the P_0, P_1, P_2 inputs respectively so that each element now shifts right when the Parallel Enable is High and left when it is Low. For left shifting, Q_0 is the serial data output and P_3 is the serial data input.

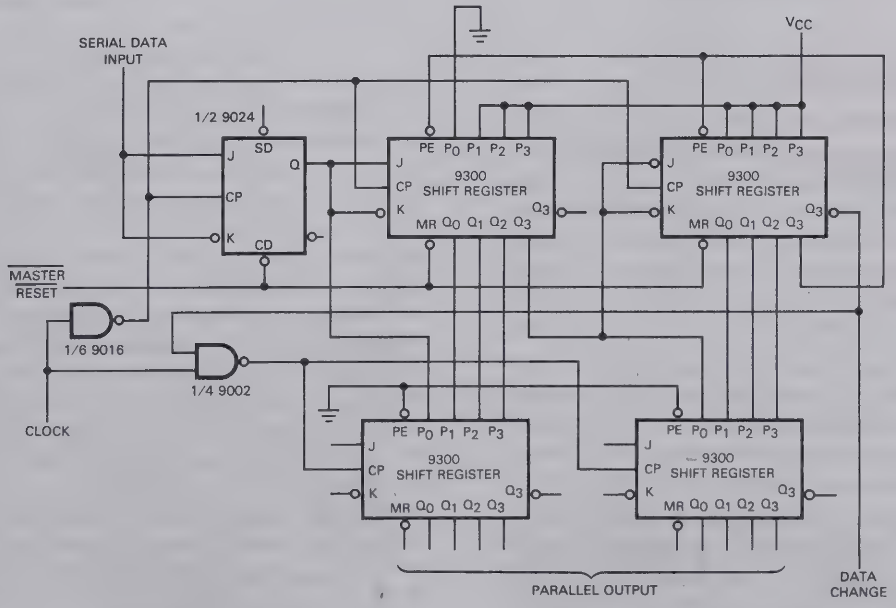
9300 SHIFT REGISTER APPLICATIONS

SERIAL-TO-PARALLEL CONVERSION


SERIAL DATA INPUT
VCC
PE P0 P1 P2 P3
J
9300 SHIFT REGISTER
CP
K
Q3
MR Q0 Q1 Q2 Q3
CLOCK
MASTER RESET
DATA OUTPUT
DATA READY

The 7-bit serial-to-parallel converter shown counts the incoming data bits with a zero marker so that a High signal results on the data ready line when the next seven bits of data are present on the seven data output lines. When the register is full, a zero marker bit has reached the last position of the register and provides a Low signal to the $\overline{PE}$ input so that on the next clock a parallel load takes place. The next data bit from the serial line plus the zero marker bit followed by ones is loaded to fill the register. The $\overline{Q}_3$ output of the last bit in the register provides a signal indicating the register is full.

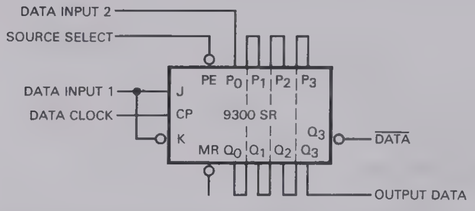
SERIAL-TO-PARALLEL CONVERSION WITH HOLDING REGISTER


SERIAL DATA INPUT
VCC
1/2 9024
SD
J
Q
CP
K
CD
PE P0 P1 P2 P3
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
MASTER RESET
1/6 9016
1/4 9002
CLOCK
PARALLEL OUTPUT
DATA CHANGE

This figure shows an 8-bit serial-to-parallel converter with two 9300s used as a holding register so that the parallel data is available for more than one bit time. The $\overline{Q}_3$ output of the last bit position of the serial-to-parallel converter gates the clock into the holding register. A High-to-Low signal change on the data change output line indicates that the next eight bits of data are present at the parallel output.

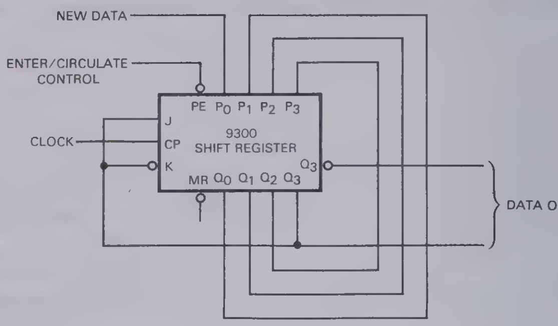
9300 SHIFT REGISTER APPLICATIONS

DUAL INPUT SHIFT REGISTER


DATA INPUT 2
SOURCE SELECT
DATA INPUT 1
DATA CLOCK
J
CP
K
PE
P0
P1
P2
P3
9300 SR
MR
Q0
Q1
Q2
Q3
Q3
DATA
OUTPUT DATA

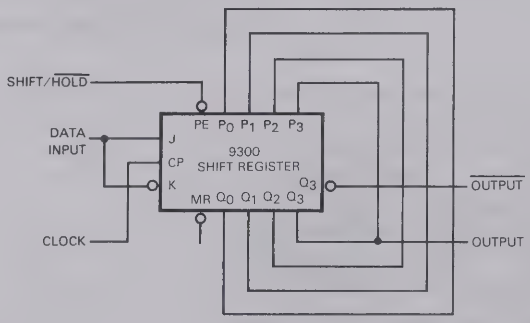
For shifting data into a register from two sources, the synchronous parallel load capability of the 9300 is useful. The circuit has a 4-bit register with the Q_0, Q_1 and Q_2 outputs connected to the P_1, P_2 and P_3 inputs so that right shifting occurs independent of the $\overline{PE}$ input. The $\overline{PE}$ input thus selects either the P_0 or the $J\overline{K}$ inputs to the register.

RECIRCULATION


NEW DATA
ENTER/CIRCULATE CONTROL
CLOCK
J
CP
K
PE
P0
P1
P2
P3
9300 SHIFT REGISTER
MR
Q0
Q1
Q2
Q3
Q3
DATA OUT

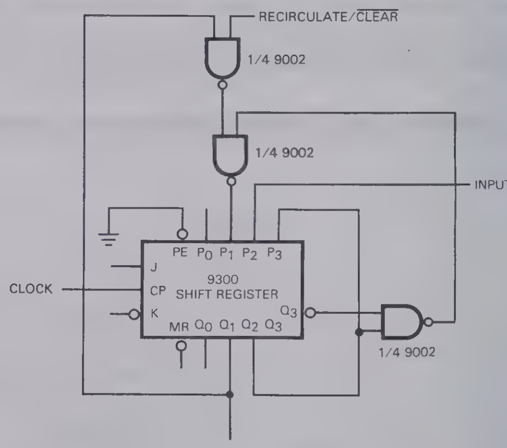
One obvious use of the 2-input shift register is to use one input for new data and the other for recirculation with the Parallel Enable input acting as an enter/circulate control as shown. With these connections, four clock periods are required to recirculate data.

SHIFT/HOLD CAPABILITY


SHIFT/HOLD
DATA INPUT
CLOCK
J
CP
K
PE
P0
P1
P2
P3
9300 SHIFT REGISTER
MR
Q0
Q1
Q2
Q3
Q3
$\overline{OUTPUT}$
OUTPUT

The synchronous parallel load also can provide a shift enable capability by connecting the register outputs to the same weighted parallel input as above. With the Parallel Enable High, information normally shifts right. When the Parallel Enable is Low, register data is reloaded continuously on each clock to hold data. The 93H72 is better suited for this application since it can hold data without sacrificing the parallel load capability.

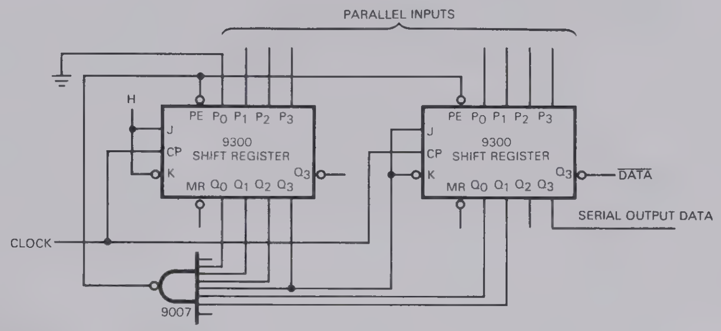
EDGE DETECTOR WITH STORAGE


RECIRCULATE/CLEAR
1/4 9002
1/4 9002
INPUT
CLOCK
J
CP
K
PE
P0
P1
P2
P3
9300 SHIFT REGISTER
MR
Q0
Q1
Q2
Q3
Q3
1/4 9002

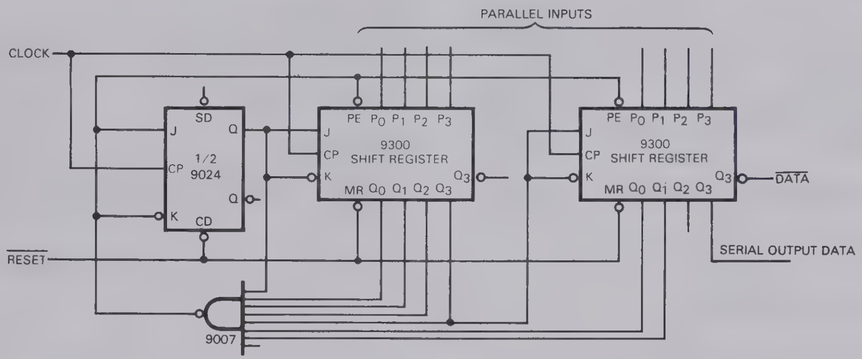
In this circuit, the four D flip-flops obtained by operating the 9300 in the parallel mode form an edge detector plus a circulating storage bit. The positive transition of the input is detected and a one is stored in the recirculating storage bit as a flag bit to indicate that the edge has occurred. The edge is detected by decoding the information in the 2-bit shift register Q_2, Q_3. The register holds input information during two clock periods so that when a Low-High sequence appears (corresponding to an edge) a one is inserted into the bit of recirculating storage (Q_1). The one will recirculate until the clear line is brought Low. The unused first flip-flop could serve as a storage bit for the carry of an arithmetic operation or as an extra bit of shift register if the clock and reset signals are compatible.

9300 SHIFT REGISTER APPLICATIONS

PARALLEL-TO-SERIAL CONVERSION


PARALLEL INPUTS
H
PE P_0 P_1 P_2 P_3
J
CP
K
9300
SHIFT REGISTER
Q_3
MR Q_0 Q_1 Q_2 Q_3
PE P_0 P_1 P_2 P_3
J
CP
K
9300
SHIFT REGISTER
Q_3
MR Q_0 Q_1 Q_2 Q_3
$\overline{DATA}$
SERIAL OUTPUT DATA
CLOCK
9007

a.


PARALLEL INPUTS
CLOCK
SD
J
Q
1/2
9024
CP
$\overline{Q}$
K
CD
PE P_0 P_1 P_2 P_3
J
CP
K
9300
SHIFT REGISTER
Q_3
MR Q_0 Q_1 Q_2 Q_3
PE P_0 P_1 P_2 P_3
J
CP
K
9300
SHIFT REGISTER
Q_3
MR Q_0 Q_1 Q_2 Q_3
$\overline{DATA}$
SERIAL OUTPUT DATA
$\overline{RESET}$
9007

b.

Figures a and *b* above show two parallel-to-serial conversion circuits. The circuit on the top is a 7-bit converter which illustrates the approach used when the number of bits to be converted is one less than an integral multiple of four bits. The circuit on the bottom has an additional flip-flop to produce an 8-bit converter. One additional gate is used in both circuits to time the conversions so that a marker bit can be used to determine when the next parallel input must be loaded. The parallel load enable ($\overline{PE}$) input to the shift register(s) is activated in both circuits when the shift register contains ones in all but the last two positions. At this time, the last position contains the last data bit for serial output from the previous parallel load, and the next-to-last bit contains a 0. Until this 0 was shifted into the next-to-last position, it held the $\overline{PE}$ input High during the shifting out of the data. The output of the gate driving the parallel load enable now goes Low so that on the next clock the next parallel data word will be loaded along with a new zero marker bit.

This converter uses a zero marker bit to count the data bits shifted out. After the parallel load, (which loads a zero followed by seven data bits), each shift brings in a one. When the register contains six adjacent ones, the output of the last stage is the last data bit from the previous load. At this time, the gate output becomes Low, enabling a load on the next clock.

The 8-bit shift register shown uses the two 9300 shift registers and half of a 9024 flip-flop. When the first seven bits of this register are ones the 9007 gate is enabled. The output of this gate enables the Parallel Load of the 9300s and supplies a zero input to the flip-flop so that on the next clock the register is loaded with a zero followed by eight bits of data. Seven clocks after the parallel load, the last data bit is in the last position.

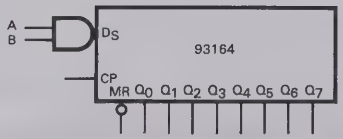
The Fairchild 9300 4-bit shift register is a general purpose sequential building block not limited to register operation but useful in a wide variety of sequential digital applications. It can be used in many counting circuits including simple counters, variable modulo counters, up/down counters and increment/decrement counters. These applications are illustrated and described in counter applications. The 9300 can be used for BCD-to-binary and binary-to-BCD conversion as described in operators.

93164 8-BIT SERIAL TO PARALLEL CONVERTER

DESCRIPTION AND OPERATION

LEADS		LOADING
A, B	Serial Data Inputs	1 UL
$\overline{MR}$	Master Reset (Active Low) Input	1 UL
$Q_0 - Q_7$	Parallel Outputs	5 UL
CP	Clock (Active High Going Edge) Input	1 UL

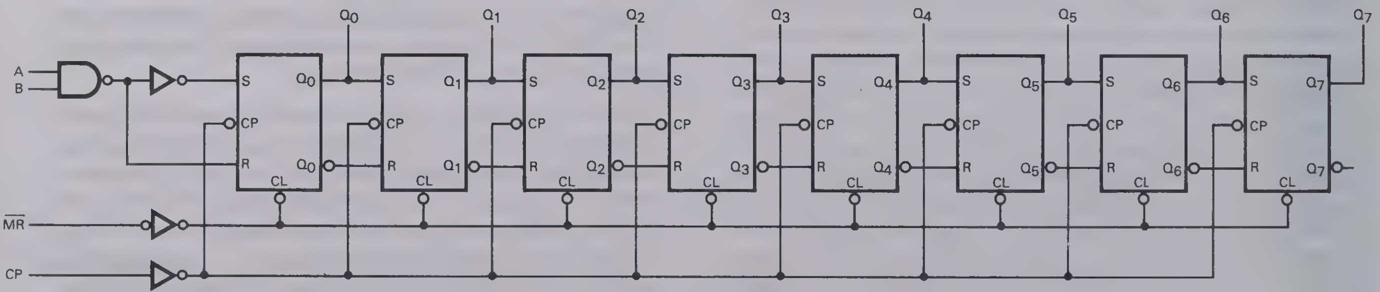
CHARACTERISTICS	
TYPICAL SPEED	36 MHz
TYPICAL DELAY	20 ns
PACKAGE	14-lead DIP or Flatpak
TYPICAL POWER DISSIPATION	180 mW



t_n		t_{n+1}
A	B	Q_0
L	L	L
L	H	L
H	L	L
H	H	H

t_n = present state

t_{n+1} = state after next clock



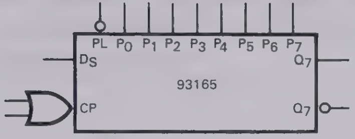
The 93164 is an 8-bit shift register with serial data entry. The output of each stage is available. A Master Reset is also provided for asynchronously clearing the register. Data enters serially through one of two inputs (A or B). The unused line must be held High or both lines should be connected together. Each Low-to-High clock transition shifts the data one place to the right. Master Reset ($\overline{MR}$) Low overrides all other inputs and asynchronously clears the register.

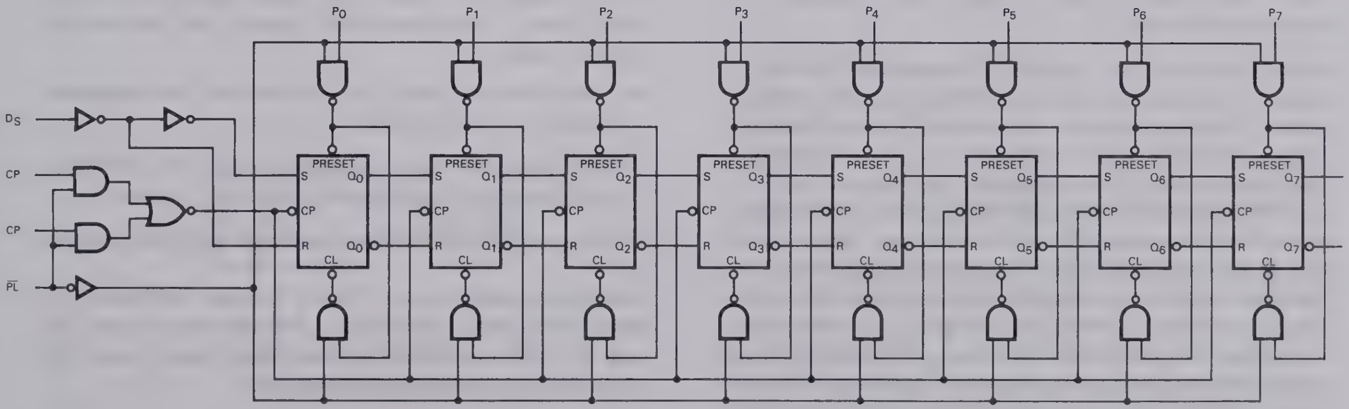
93165 8-BIT PARALLEL TO SERIAL CONVERTER

DESCRIPTION AND OPERATION

LEADS		LOADING
$\overline{PL}$	Parallel Load (Active Low) Input	2 UL
$P_0 - P_7$	Parallel Inputs	1 UL
D_S	Serial Data Input	1 UL
CP	Clock (Active High Going Edge) Inputs	1 UL
Q_7	Serial Output from Last Stage	10 UL
$\overline{Q}_7$	Complementary Output	10 UL

CHARACTERISTICS	
TYPICAL SPEED	26 MHz
TYPICAL DELAY	20 ns
PACKAGE	16-lead DIP or Flatpak
TYPICAL POWER DISSIPATION	230 mW

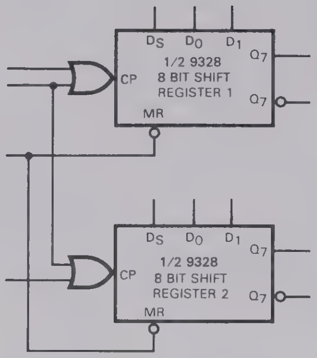




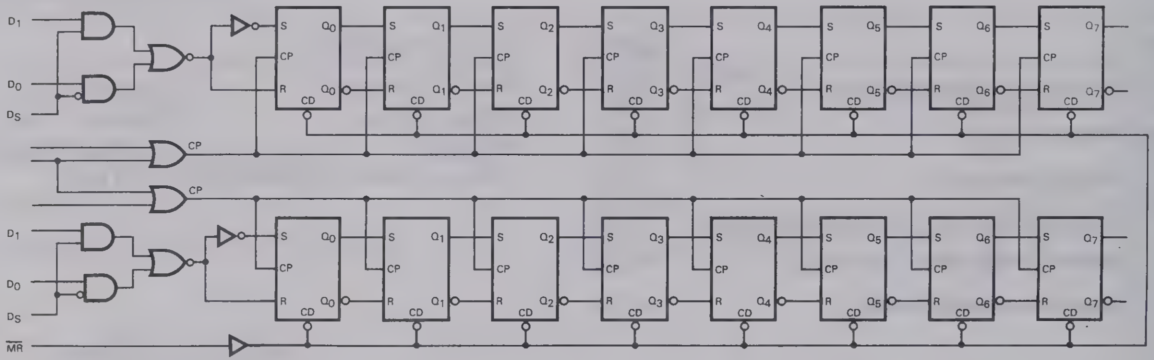
The 93165 is an 8-Bit parallel load or serial-in shift register with Q_7 and $\overline{Q}_7$ outputs available from the last stage. Parallel loading is accomplished asynchronously when the Parallel Load ($\overline{PL}$) input is brought Low. With the $\overline{PL}$ input in the High state, data enters the register serially at the D input and is shifted one place to the right with each positive-going clock transition. The clock has two inputs; either can be used as the actual clock input. The other input can then be used as a clock inhibit by holding it High.

9328 DUAL 8-BIT SHIFT REGISTER

DESCRIPTION AND OPERATION



LEADS		LOADING
D_S	Data Select Input	2.0 UL
D_0, D_1	Data Inputs	1.0 UL
CP	Clock (Active High Going Edge) Input	
	Common	3.0 UL
	Separate	1.5 UL
$\overline{MR}$	Master Reset (Active Low) Input	1.0 UL
Q_7	Last Stage Output	10.0 UL
$\overline{Q}_7$	Complementary Output	10.0 UL



The 9328 dual 8-bit shift register features active pull up outputs, high speed, excellent noise margins and input clamp diodes that minimize the adverse effects of line reflections.

The 9328 dual shift register includes two groups of eight clocked RS master/slave flip-flops and several gates. The logic symbol and loading rules are shown above. The two shift registers have both a common clock input (lead 9) and separate clock inputs (leads 7 and 10). The clock OR gate drives the eight clock inputs of the flip-flop in parallel. When the two clock inputs (the separate and the common) to the OR gate are Low, the slave latches are inactive, but data can enter the master latches via the R and S inputs. If either clock input goes High with the other Low, or if both go High simultaneously, two things happen. Data inputs R and S are inhibited so a subsequent change in input data does not affect the master, and the information in the master is transferred to the slave. When the transfer is complete, both the master and the slave remain steady as long as at least one clock input remains High.

During the High-to-Low transition of this last High clock input, the transfer path from the master to the slave is inhibited so the slave does not change its state. Also, data inputs R and S are enabled and new data can enter the master. Either clock input can act as a clock enable input and a logic Low signal enables the other clock input. The three clock inputs can be used in several ways:

- One common clock for 16 bits and two separate clocks for eight bits each.
- One common clock for 16 bits and two separate clock enables, one for each 8-bit shift register.
- Two separate clocks and one common clock enable.

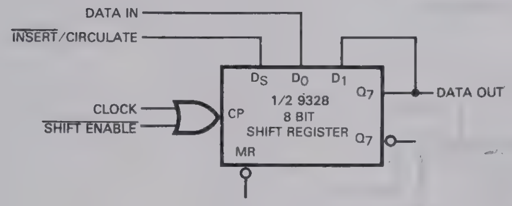
Each 8-bit shift register has a 2-input multiplexer in front of the serial data input. The two data inputs, D_0 and D_1 are controlled by the Data Select input D_S according to the Boolean expression:

$$\text{Serial Data In: } S_D = \overline{D}_S D_0 + D_S D_1$$

Both assertion and negation outputs of each 8-bit shift register are available for use. An asynchronous Master Reset ($\overline{MR}$) with a Low logic level clears all 16 stages regardless of other inputs.

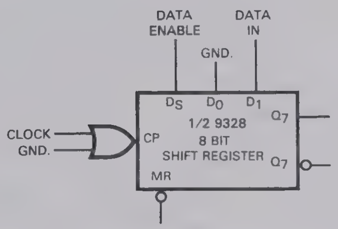
9328 DUAL 8-BIT SHIFT REGISTER

RECIRCULATION


DATA IN
INSERT/CIRCULATE
CLOCK
SHIFT ENABLE
DS
D0
D1
Q7
1/2 9328
8 BIT
SHIFT REGISTER
CP
MR
DATA OUT

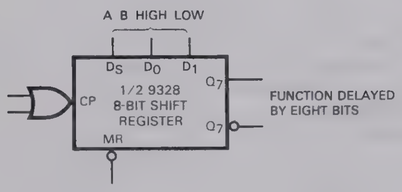
The 9328 provides economical high speed serial storage. It can accept data from any of two data sources via the 2-input multiplexers at the input of each 8-bit shift register. This figure illustrates how this capability is frequently used. With the setup shown, information continues to circulate through the feedback connection and new data is inserted at the required time by the Data Select control.

DATA ENABLE


DATA ENABLE
DATA IN
GND.
DS
D0
D1
Q7
1/2 9328
8 BIT
SHIFT REGISTER
CP
MR
CLOCK
GND.

The input multiplexer of the 9328 can be used as a data enable circuit, shifting zeros into the register whenever the Data Select input is Low.

MINIMUM INPUT PATTERN/TWO VARIABLES

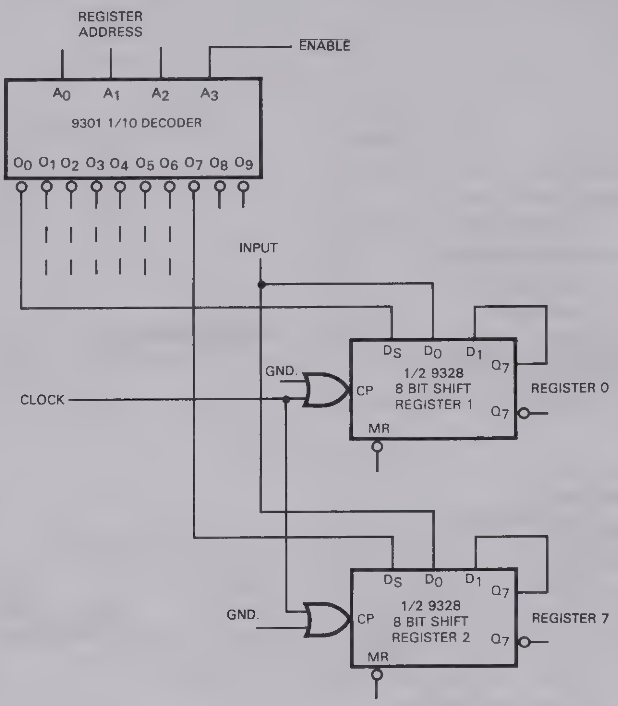

A B HIGH LOW
DS
D0
D1
Q7
CP
1/2 9328
8-BIT SHIFT
REGISTER
MR
FUNCTION DELAYED
BY EIGHT BITS

The input multiplexer of the 9328 can also be used to generate any one of the 16 theoretically possible functions of the two variables A and B. The table shows how the three inputs D_S, D_0 and D_1 must be connected for this purpose.

	CONNECTION TO			A	L	H	L	H
FUNCTION	D_S	D_0	D_1	B	L	L	H	H
All Zeros	X	L	L		L	L	L	L
$\overline{A} \bullet \overline{B} = \overline{A + B}$	A	$\overline{B}$	L		H	L	L	L
$A \bullet \overline{B}$	B	A	L		L	H	L	L
$\overline{B}$	B	H	L		H	H	L	L
$\overline{A} \bullet B$	A	B	L		L	L	H	L
$\overline{A}$	A	H	L		H	L	H	L
$A \oplus B$	A	B	$\overline{B}$		L	H	H	L
$\overline{A} + \overline{B} = \overline{A \bullet B}$	A	H	$\overline{B}$		H	H	H	L
$A \bullet B$	A	L	B		L	L	L	H
$\overline{A \oplus B}$	A	$\overline{B}$	B		H	L	L	H
A	A	L	H		L	H	L	H
$A + \overline{B}$	B	H	A		H	H	L	H
B	B	L	H		L	L	H	H
$\overline{A} + B$	A	H	B		H	L	H	H
$A + B$	A	B	H		L	H	H	H
All Ones	X	H	H		H	H	H	H

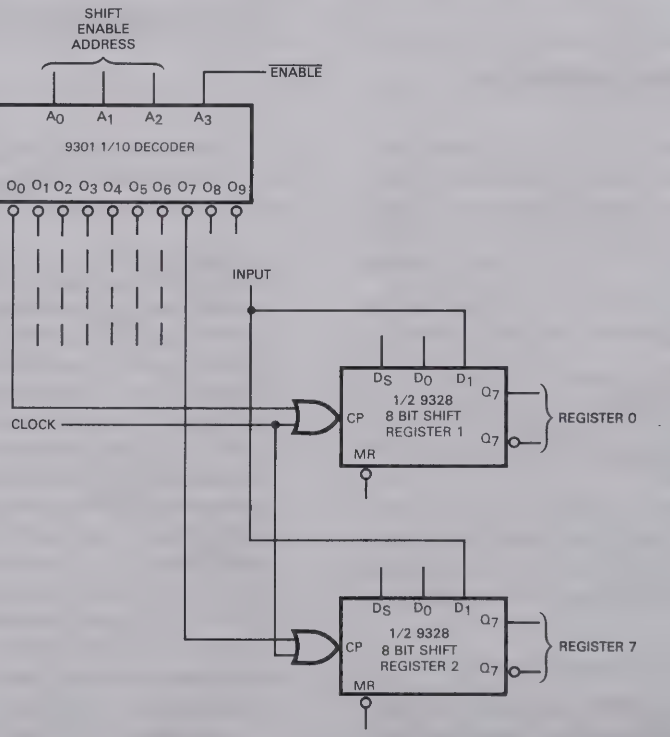
9328 DUAL 8-BIT SHIFT REGISTER

DATA INPUT SELECTION


REGISTER ADDRESS
ENABLE
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
INPUT
GND.
CLOCK
CP
DS D0 D1
1/2 9328
8 BIT SHIFT
REGISTER 1
Q7
Q7
MR
REGISTER 0
GND.
1/2 9328
8 BIT SHIFT
REGISTER 2
REGISTER 7

The use of D_S, D_0, D_1 to provide enable capability can be extended with a 9301 active Low output decoder to control data flow in a bank of eight shift registers. In this circuit, data enters the selected register under control of the address of the 9301. Data recirculates in the non-selected registers.

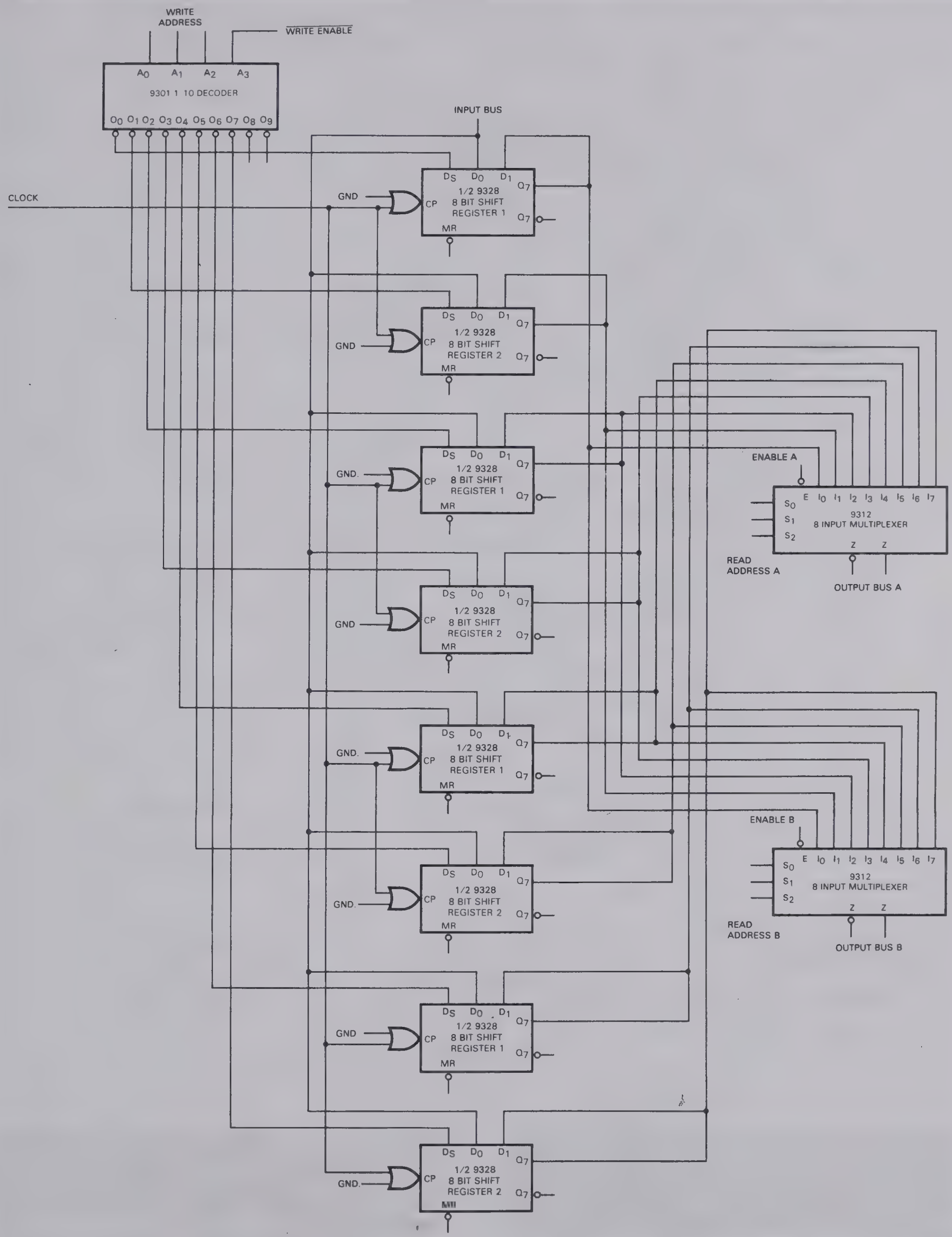
SHIFT REGISTER SELECTION


SHIFT ENABLE ADDRESS
ENABLE
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
INPUT
CLOCK
CP
DS D0 D1
1/2 9328
8 BIT SHIFT
REGISTER 1
Q7
Q7
MR
REGISTER 0
1/2 9328
8 BIT SHIFT
REGISTER 2
REGISTER 7

One of the clock inputs on the shift register can serve as an active Low shift select control. The 9301 determines which shift register from a bank of shift registers can accept data inputs. The input clock loading of the 9328 permits a single 9301 decoder to control 10 shift registers of up to 48 bits in length.

9328 DUAL 8-BIT SHIFT REGISTER APPLICATIONS

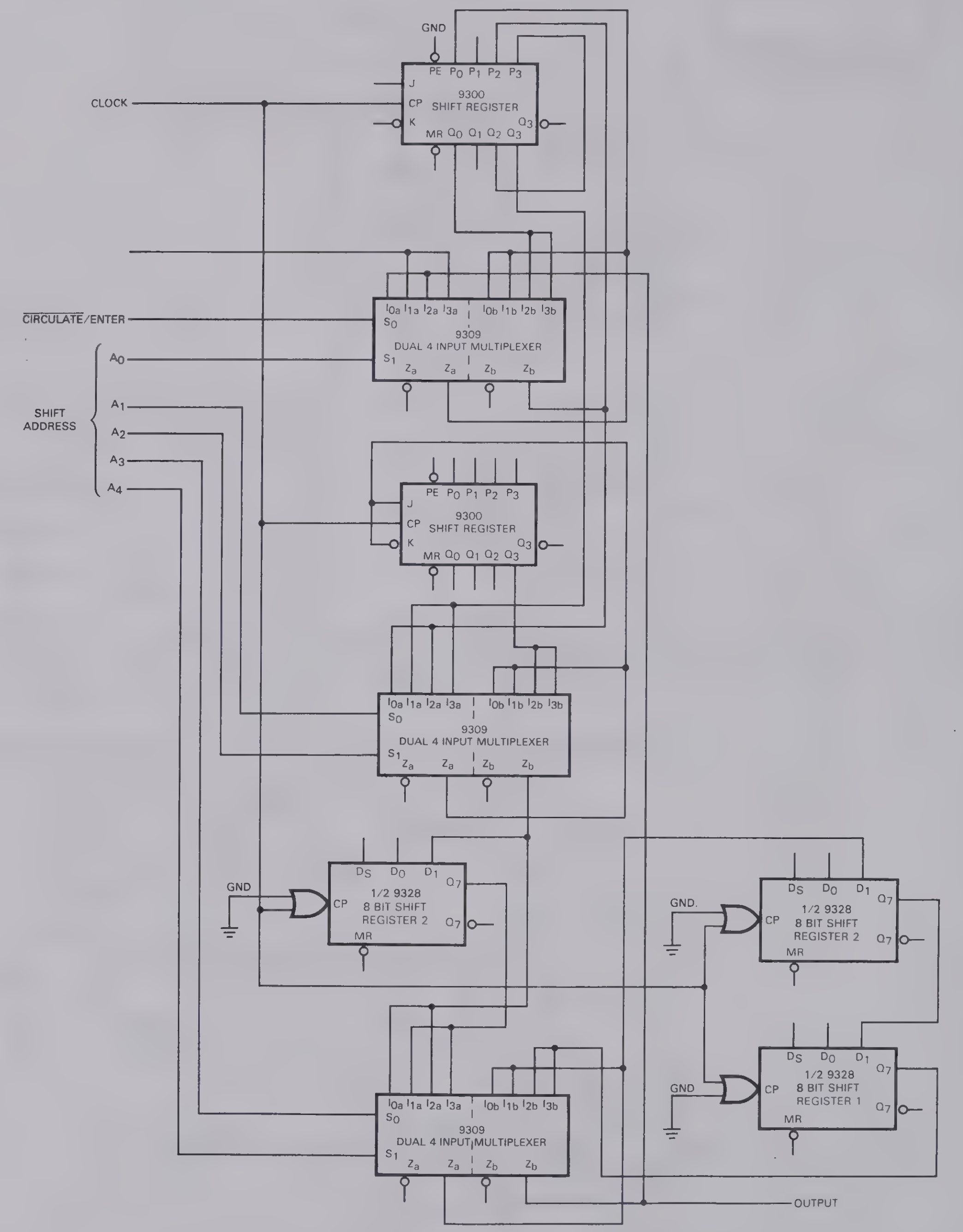
SERIAL MEMORIES


WRITE ADDRESS
WRITE ENABLE
A0 A1 A2 A3
9301 1 10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
INPUT BUS
CLOCK
GND
DS D0 D1
1/2 9328
8 BIT SHIFT
REGISTER 1
REGISTER 2
CP
MR
Q7
ENABLE A
E I0 I1 I2 I3 I4 I5 I6 I7
S0 S1 S2
9312
8 INPUT MULTIPLEXER
Z Z
READ ADDRESS A
OUTPUT BUS A
ENABLE B
READ ADDRESS B
OUTPUT BUS B

Shown is a circuit diagram of an 8-word by 8-bit serial/parallel memory using eight 9328s. In this circuit, serial data enters any of the four 9328s. A 9301 1-of-10 decoder determines which register the data enters and whether the data is old or new. The eight outputs of the parallel shift registers are multiplexed by two 8-input 9312s providing two serial independent output data streams, A and B. These data streams are then applied to additional serial processing logic.

9328 DUAL 8-BIT SHIFT REGISTER APPLICATIONS

VARIABLE LENGTH SHIFT REGISTER


GND
PE P0 P1 P2 P3
J
CP
K
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
CLOCK
CIRCULATE/ENTER
SHIFT ADDRESS
A0
A1
A2
A3
A4
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309 DUAL 4 INPUT MULTIPLEXER
Za Za Zb Zb
DS D0 D1
Q7
1/2 9328 8 BIT SHIFT REGISTER 2
1/2 9328 8 BIT SHIFT REGISTER 1
MR
GND.
OUTPUT

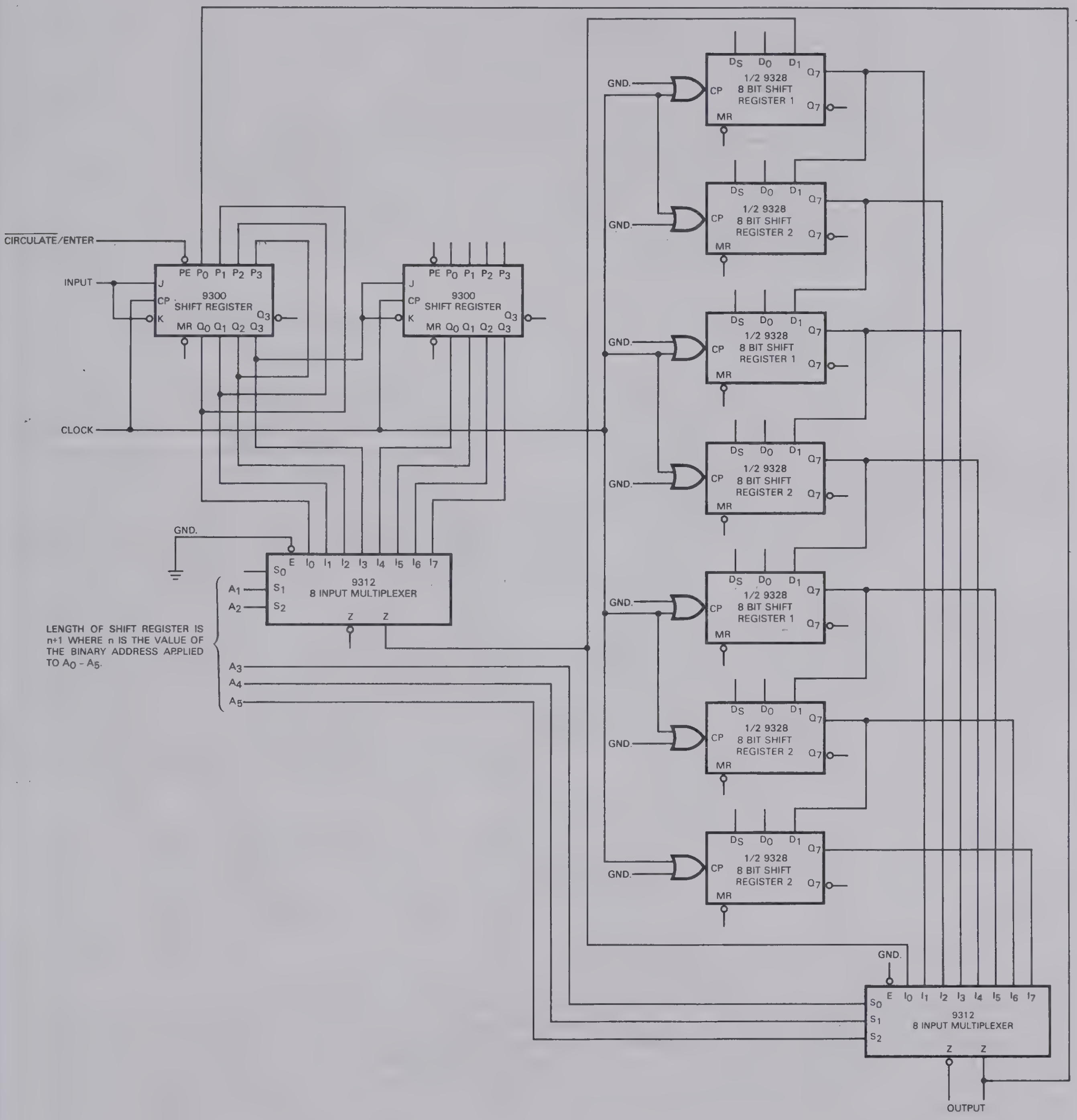
The 9328 can be used with other MSI devices as a variable-length shift register. This example uses three 9328s, three 9309 dual 4-input multiplexers and two 9300 shift registers. This shift register can shift data to 31 locations with appropriate logic to the shift address inputs.

Data enters the first half of one of the 9309 multiplexers, governed by a circulate/enter control. The output of this half of the 9309 then enters the second half in one of two ways: directly, if the required length register is even; or via a single shift delay if the required length register is odd. Then, the output of the second half of the 9309 multiplexer is fed directly to another 9309 or via a 2-bit shift register if a shift of two places is required.

This procedure continues with each half of each multiplexer switching the data either directly from a previous stage or via a binary-weighted shift to its output. The final output is fed back to the first multiplexer to provide circulation capability.

9328 DUAL 8-BIT SHIFT REGISTER APPLICATIONS

VARIABLE LENGTH SHIFT REGISTER

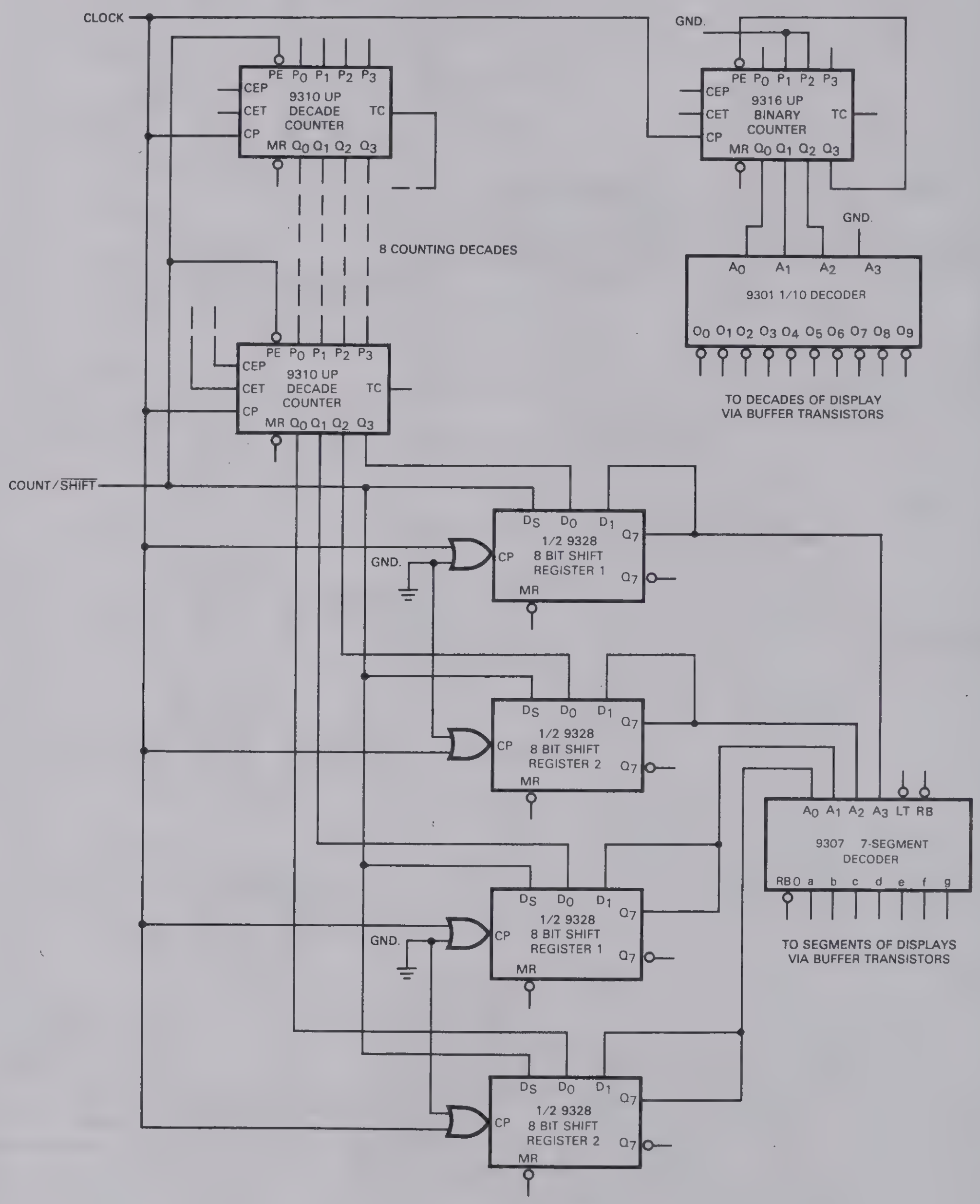

CIRCULATE/ENTER
INPUT
CLOCK
PE P0 P1 P2 P3
J
CP
K
9300 SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
GND.
E I0 I1 I2 I3 I4 I5 I6 I7
S0
S1
S2
A1
A2
9312 8 INPUT MULTIPLEXER
Z Z
A3
A4
A5
DS D0 D1
Q7
1/2 9328 8 BIT SHIFT REGISTER 1
1/2 9328 8 BIT SHIFT REGISTER 2
MR
OUTPUT

LENGTH OF SHIFT REGISTER IS n+1 WHERE n IS THE VALUE OF THE BINARY ADDRESS APPLIED TO A0 - A5.

A variable-length shift register with slightly different wiring is illustrated above. In this circuit, the first 9300 permits two lots of data to enter: (1) the input information to the variable-length shift register, and (2) the variable-length shift register's final output (which recirculates through the circuit). To provide for these two inputs, the device is connected as a shift register in the parallel mode and the Parallel Enable line is used as a circulate/enter control. The second 9300 does not have this dual entry mode of operation. These two 9300s constitute an 8-bit shift register. A 9312 8-input multiplexer switches the outputs of the two 9300s to provide variable-length shifting up to eight stages. The output of the 9312 is then fed to one of the eight inputs of a second 9312. The seven other inputs come from three and one-half 9328s which provide shifting in increments of eight bits. The address inputs of the two 9312 multiplexers can shift the incoming re- the required number of places, up to a maximum of 64 places.

9328 DUAL 8-BIT SHIFT REGISTER APPLICATIONS

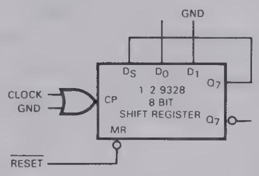
MULTIPLEXED 7-SEGMENT COUNT & DISPLAY CIRCUIT


CLOCK
COUNT/$\overline{\text{SHIFT}}$
9310 UP DECADE COUNTER
CEP
CET
CP
PE
P0 P1 P2 P3
TC
MR Q0 Q1 Q2 Q3
8 COUNTING DECADES
GND.
9316 UP BINARY COUNTER
9301 1/10 DECODER
A0 A1 A2 A3
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
TO DECADES OF DISPLAY VIA BUFFER TRANSISTORS
DS D0 D1
1/2 9328 8 BIT SHIFT REGISTER 1
1/2 9328 8 BIT SHIFT REGISTER 2
Q7
$\overline{\text{Q7}}$
MR
9307 7-SEGMENT DECODER
A0 A1 A2 A3 LT RB
RBO a b c d e f g
TO SEGMENTS OF DISPLAYS VIA BUFFER TRANSISTORS

A counting multiplex display scheme for controlling 7-segment numeric displays is shown here. The two 9328 shift registers hold eight decades of information and each decade sequentially addresses a 9307 decoder which drives the segments of 7-segment displays via buffer transistors. The 9310 decade counters count to the desired value and when display is required, the recirculation loop is broken and the eight new decades of information are shifted into the shift registers. At any clock time, the character to be displayed is controlled by two devices: a 9316 counter connected to divide by eight and a decoder with eight mutually exclusive control lines. These control lines are usually buffered by transistors for each character. Low speed industrial counter applications are covered in counters.

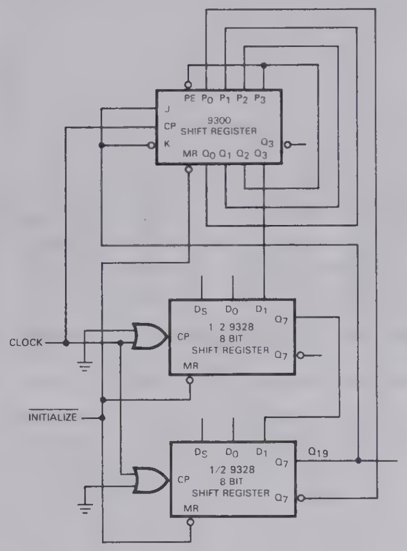
9328 DUAL 8-BIT SHIFT REGISTER APPLICATIONS

MODULO 16 COUNTER


GND
DS D0 D1
Q7
CLOCK
GND
CP
1/2 9328
8 BIT
SHIFT REGISTER
Q7
MR
RESET

The 9328 can act as a shift register counter in which several states are decoded and fed back to the input to force the register to pass through a loop of states. Only the last stage of each shift register is available so a limited number of counters can be built using one 9328. This figure shows one half of a 9328 connected as a modulo 16 counter producing a sequence of eight zeros and then eight ones. The counter must be initialized since the same logic configuration could also produce other sequences; e.g., 010101, the sequence generated by a modulo 2 counter. The 9328 can also be used in increment/decrement counting schemes for multi-decade operations. This application is covered in counter applications.

SIMPLE PSEUDO-RANDOM SEQUENCE GENERATOR

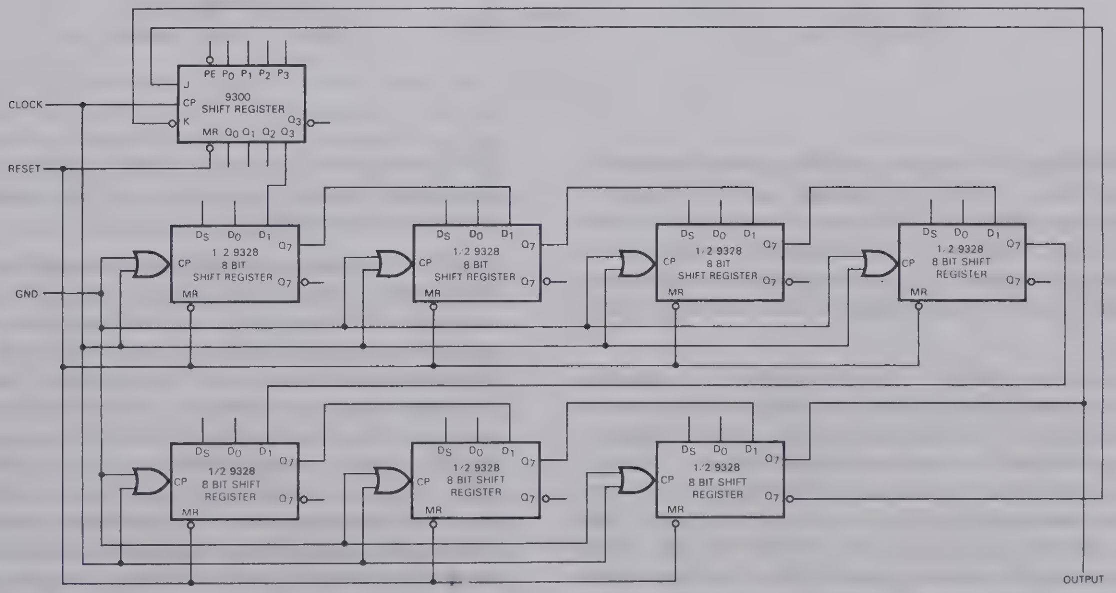

PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
CLOCK
DS D0 D1
Q7
1/2 9328
8 BIT
SHIFT REGISTER
Q7
CP
MR
INITIALIZE
DS D0 D1
Q7
Q19
1/2 9328
8 BIT
SHIFT REGISTER
Q7
CP
MR

A simple pseudo-random sequence generator is illustrated. The circuit uses only two ICs and recirculates every 50 milliseconds with a 20 MHz clock frequency. The required feedback connection can be expressed as:

$$Q_2 \odot Q_{19} = \overline{Q}_2\overline{Q}_{19} + Q_2Q_{19}.$$

To provide this logic without additional gates, Q_2 is fed into the Parallel Enable of the 9300 shift register connected to act as a shift register even when parallel loading takes place. When Q_2 is Low, the input to the shift register is $\overline{Q}_{19}$; when Q_2 is High, the input is Q_{19} via the normal $J\overline{K}$ inputs.

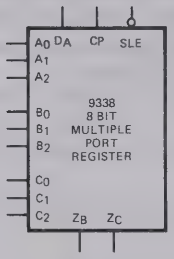
LONG PSEUDO-RANDOM SEQUENCE GENERATOR


PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
CLOCK
RESET
GND
DS D0 D1
Q7
1/2 9328
8 BIT
SHIFT REGISTER
CP
MR
1/2 9328
8 BIT SHIFT
REGISTER
OUTPUT

The 9328 can be used as part of a long shift counter to provide a pseudo-random sequence, as shown. This counter passes through 2^{60}-1 states, so many that even at a frequency of 20 million states per second the counter would not repeat until more than 18 centuries had elapsed.

9338 8-BIT MULTIPLE PORT REGISTER

DESCRIPTION AND OPERATION



LEADS		LOADING
A_0, A_1, A_2	Write Address Inputs	2/3 UL
D_A	Data Input	2/3 UL
B_0, B_1, B_2	B Read Address Inputs	2/3 UL
Z_B	B Output	10 UL
C_0, C_1, C_2	C Read Address Inputs	2/3 UL
Z_C	C Output	10 UL
CP	Clock Active High Going Edge Input	2/3 UL
$\overline{SLE}$	Slave Enable (Active Low) Input	2/3 UL

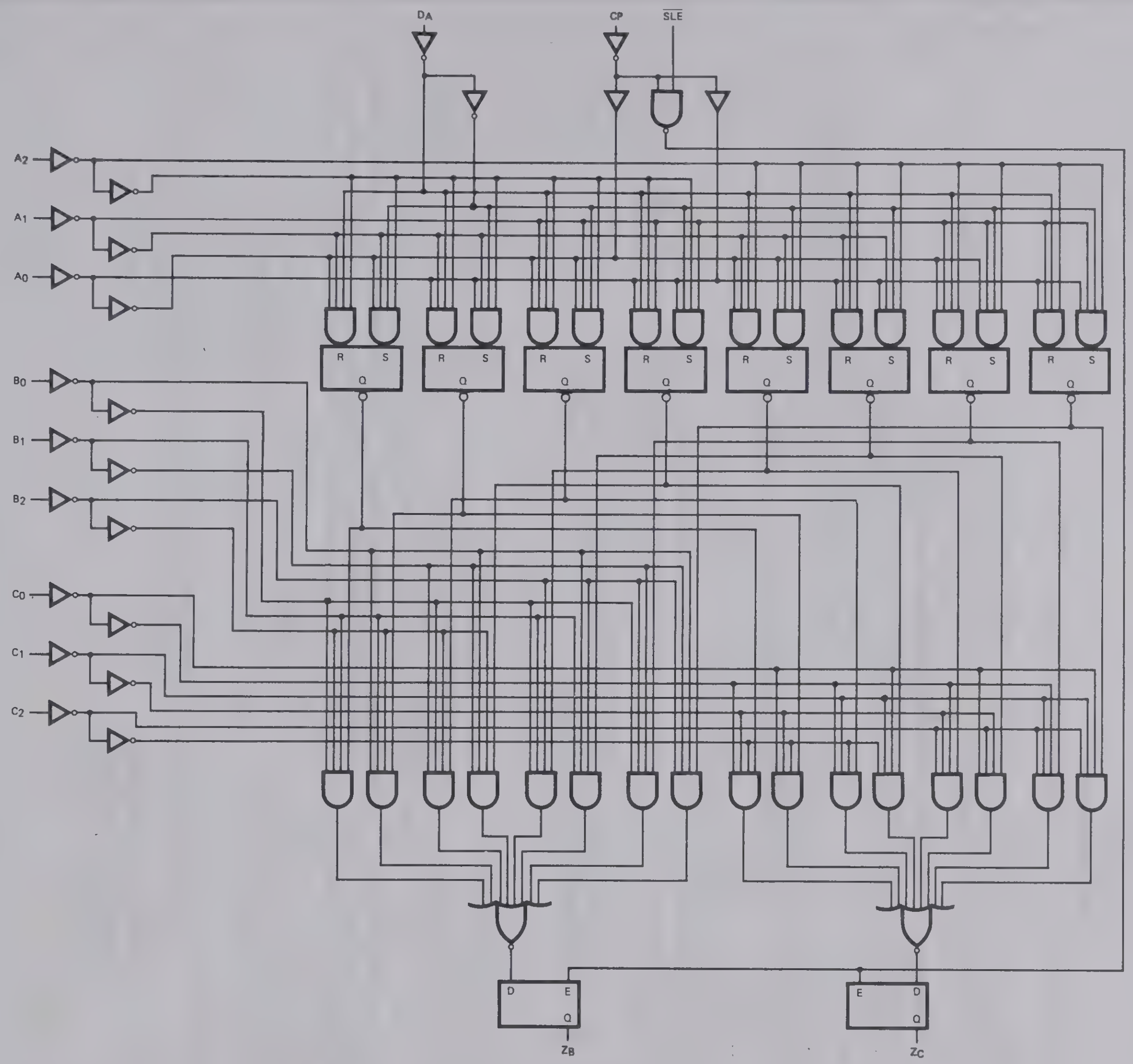
The 9338 8-bit multiple port register can be considered a 1-bit slice of eight high speed working registers. Data can be written into any one and read from any two of the eight locations simultaneously. Master/slave operation eliminates all race problems associated with simultaneous read/write activity from the same location.

Timing of data transfer is similar to that of a standard master/slave flip-flop. When the clock is Low the slaves are held steady but the information on the D (data) input is permitted to enter the selected master. The next Low-to-High clock transition locks the masters and they become insensitive to D and write address inputs. This rising clock edge also connects each of the two slaves to the selected masters, causing their contents to be reflected on the outputs. Therefore, outputs change following the Low-to-High clock transition as in most Fairchild TTL/MSI devices and TTL flip-flops. The Slave Enable ($\overline{SLE}$) input can be used to defeat the master/slave operation. If the Slave Enable line is held low ($\overline{SLE}$), the slaves are continuously enabled allowing immediate transfer of information from the selected masters to the outputs.

The 9338 features master/slave operation permitting simultaneous read/write without race problems, easy expansion for larger word size and word count, typical power dissipation of 265 mW and two output lines allowing two words to read simultaneously. The two output lines mean both inputs to an arithmetic or logical operation may be provided from one register bank of 9338s. Other characterisitics of the 9338 include high speed with typical delay from clock pulse to output of 35 nanoseconds, TTL integrated circuitry with active pull up providing high speed with reasonable power consumption and excellent noise margins, input clamp diodes to ground minimizing the effects of line reflections and input/output characteristics that provide easy interface with all Fairchild TTL and DTL devices.

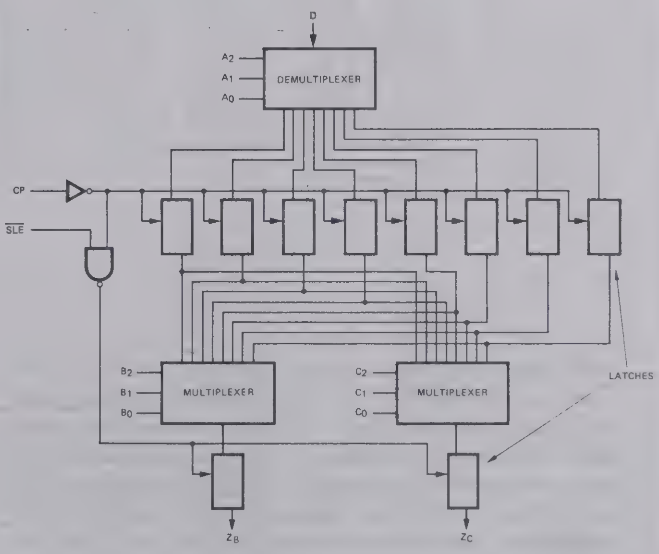
9338 8-BIT MULTIPLE PORT REGISTER

LOGIC DIAGRAM/FUNCTIONAL EQUIVALENT



The logic diagram for the 9338 is shown above. When the clock input (CP) is Low data applied to the data input line (D_A) enters the selected master. This selection is accomplished by coding the three write input select lines (A_0,A_1,A_2) appropriately. Data is stored synchronously with the rising edge of the clock pulse.

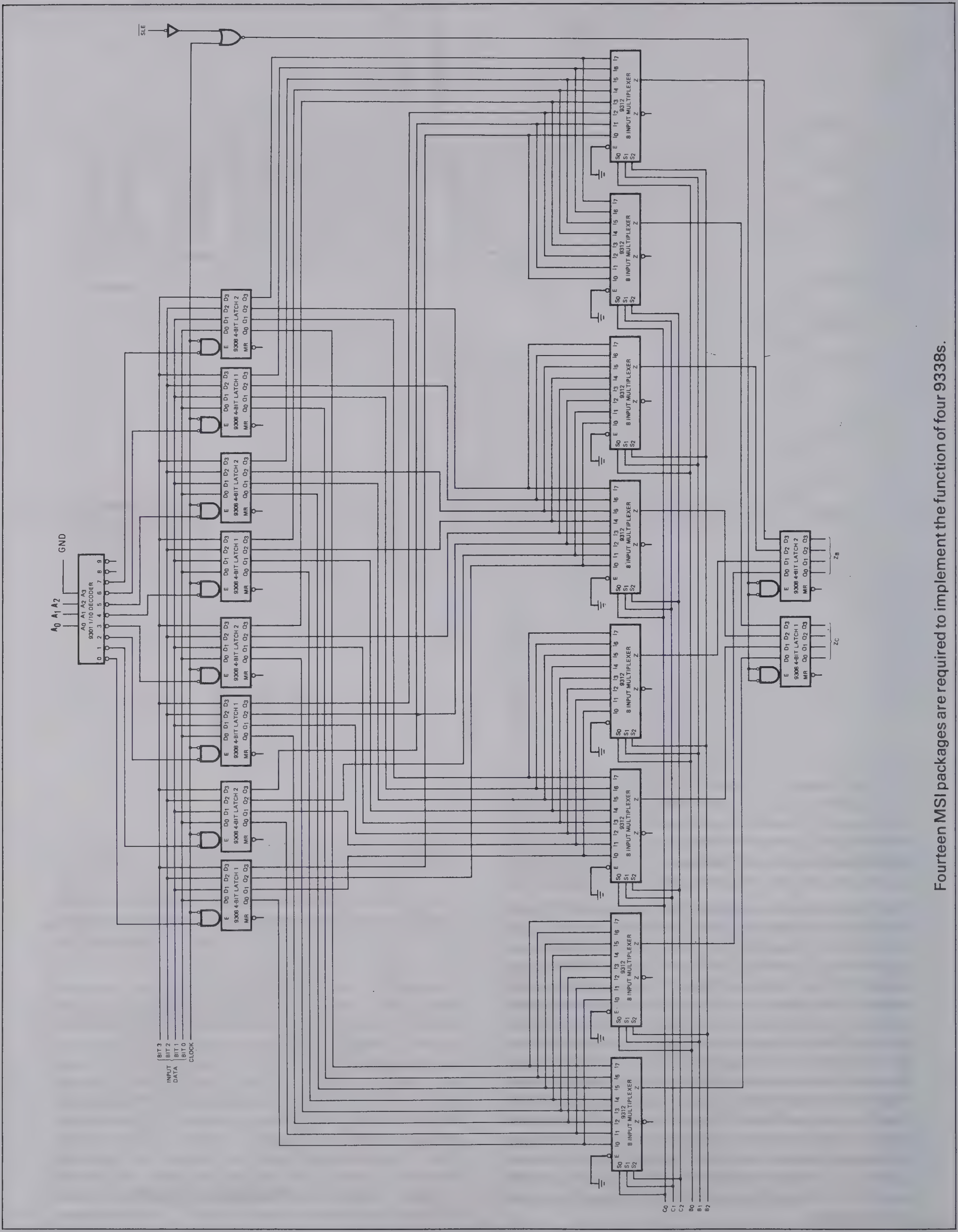
The information for each of the two slaved (output) latches is selected by two sets of read address inputs (B_0, B_1, B_2 and C_0, C_1, C_2). The information enters the slave while the clock is High and is stored while the clock is Low. If Slave Enable is Low ($\overline{SLE}$), the slave latches are continuously enabled. The signals are available on the output leads (Z_B and Z_C). The input bit selection and the two output bit selections can be accomplished independently or simultaneously. The data flows into the device, is demultiplexed according to the state of the write address lines and is clocked into the selected latch. The eight latches function as masters and store the input data. The two output latches are slaves and hold the data during the read operation. The state of each slave is determined by the state of the master selected by its associated set of read address inputs.



FUNCTIONAL EQUIVALENT

9338 8-BIT MULTIPLE PORT REGISTER

EQUIVALENT IMPLEMENTATION OF FOUR 9338s

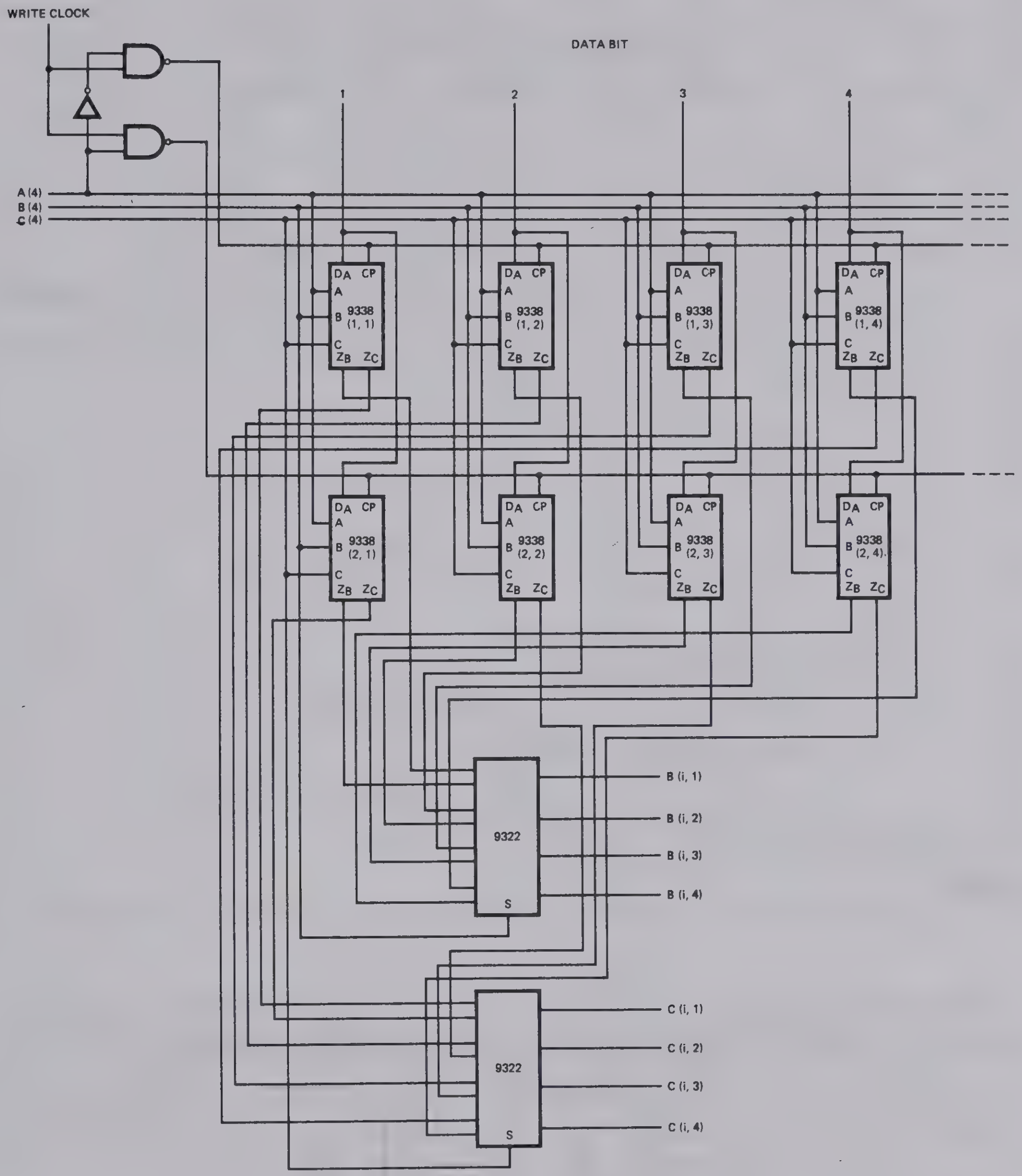


SLE
GND
A0 A1 A2
A0 A1 A2 A3
9301 1/10 DECODER
0 1 2 3 4 5 6 7 8 9
E
D0 D1 D2 D3
Q0 Q1 Q2 Q3
MR
9308 4-BIT LATCH 1
9308 4-BIT LATCH 2
I0 I1 I2 I3 I4 I5 I6 I7
S0 S1 S2
Z
9312 8 INPUT MULTIPLEXER
INPUT DATA
BIT 3
BIT 2
BIT 1
BIT 0
CLOCK
C0 C1 C2 B0 B1 B2
ZB
ZC

Fourteen MSI packages are required to implement the function of four 9338s.

9338 8-BIT MULTIPLE PORT REGISTER

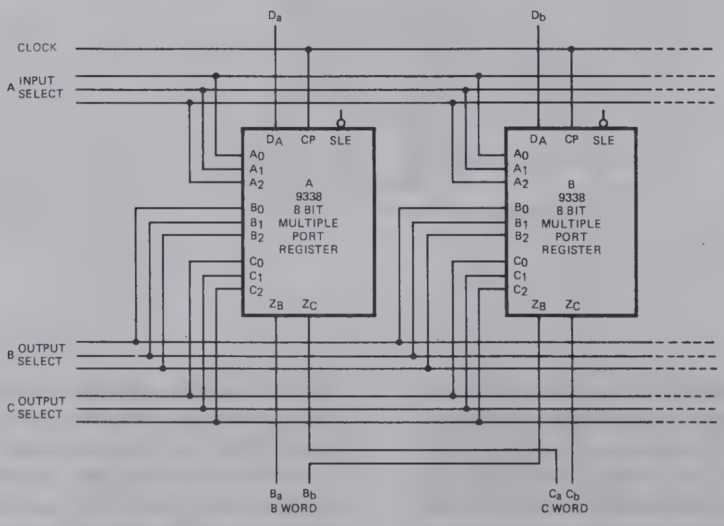
SERIAL EXPANSION CAPABILITIES


WRITE CLOCK
DATA BIT
1
2
3
4
A (4)
B (4)
C (4)
DA CP A B C ZB ZC
9338 (1, 1)
9338 (1, 2)
9338 (1, 3)
9338 (1, 4)
9338 (2, 1)
9338 (2, 2)
9338 (2, 3)
9338 (2, 4)
9322
S
B (i, 1)
B (i, 2)
B (i, 3)
B (i, 4)
C (i, 1)
C (i, 2)
C (i, 3)
C (i, 4)

9338s can be extended serially for m banks of n bits each by designing several n stage registers and multiplexing the outputs. Two banks of 16 registers are shown. Additional banks are possible by extending the addressing scheme using larger multiplexers and using a decoder to drive the clock inputs. Information is written into the required register bank by pulsing the clock line. The master/slave arrangement functions properly with this clocking scheme but is defeated by tying Slave Enable Low.

The A, B and C address lines are each four lines. The three low-order bits (A_{0-2}, B_{0-2}, C_{0-2}) are common to all 9338s and organized as in the next figure. The high order bits (A_3, B_3, C_3) select the bank for read or write operation. Bit A_3 gates the clock pulse to one of the banks. This pulse loads all of the data bits into the bank selected by A_3 and the position determined A_{0-2}. Bits B_3 and C_3 select the source of the information to be passed through the multiplexers.

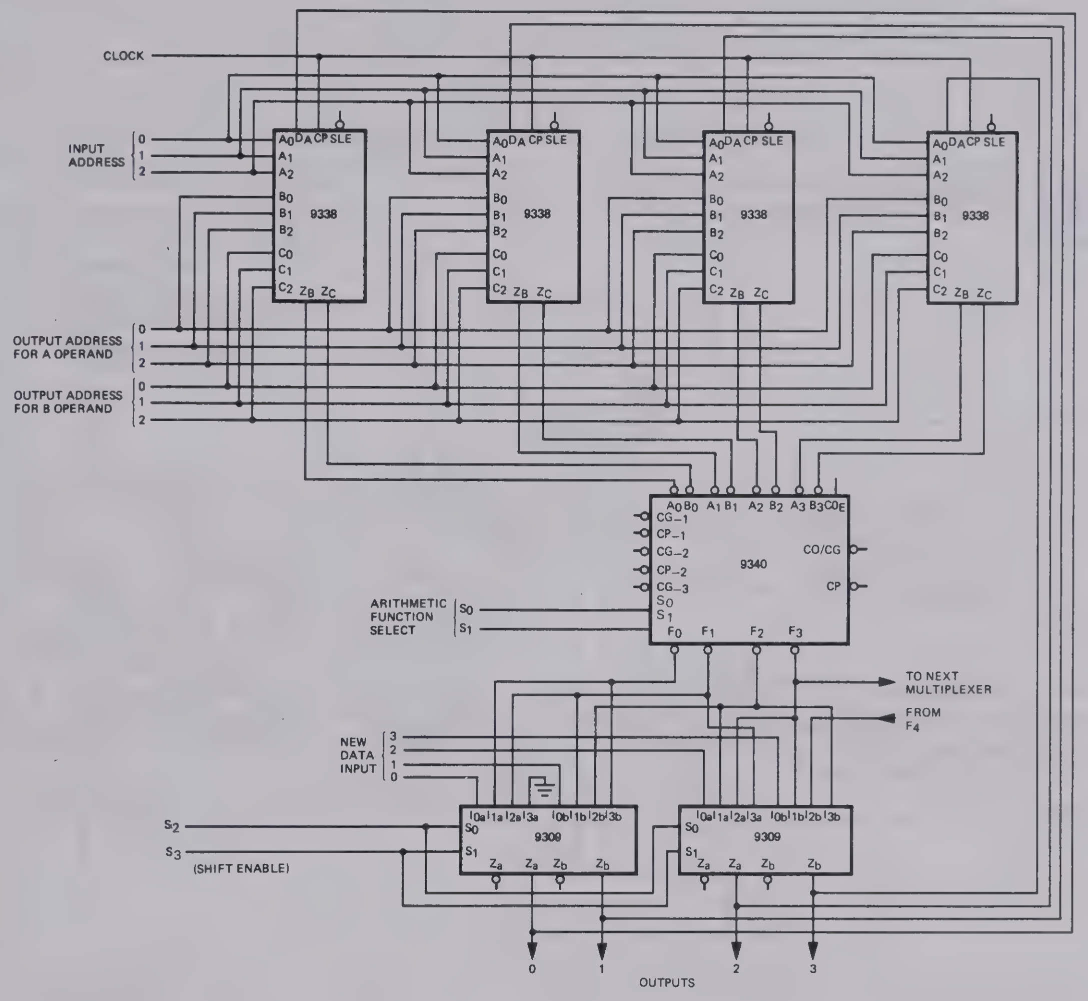
PARALLEL EXPANSION


Da
Db
CLOCK
A INPUT SELECT
DA
CP
SLE
A0
A1
A2
B0
B1
B2
C0
C1
C2
ZB
ZC
A
9338
8 BIT
MULTIPLE
PORT
REGISTER
B
9338
8 BIT
MULTIPLE
PORT
REGISTER
B OUTPUT SELECT
C OUTPUT SELECT
Ba Bb
B WORD
Ca Cb
C WORD

For parallel expansion, one 9338 is needed for each bit of required word length. The clock and address input lines should be connected in common on all of the devices. This register configuration provides two words of n bits each at one time as illustrated, where n devices are connected in parallel.

9338 8-BIT MULTIPLE PORT REGISTER APPLICATIONS

ARITHMETIC LOGIC UNIT


CLOCK
INPUT ADDRESS
0
1
2
A0 DA CP SLE
A1
A2
B0
B1
B2
C0
C1
C2
ZB ZC
9338
OUTPUT ADDRESS FOR A OPERAND
OUTPUT ADDRESS FOR B OPERAND
A0 B0 A1 B1 A2 B2 A3 B3 C0 E
CG−1
CP−1
CG−2
CP−2
CG−3
S0
S1
F0 F1 F2 F3
CO/CG
CP
9340
ARITHMETIC FUNCTION SELECT
TO NEXT MULTIPLEXER
FROM F4
NEW DATA INPUT
3
2
1
0
S2
S3
(SHIFT ENABLE)
I0a I1a I2a I3a I0b I1b I2b I3b
Za Za Zb Zb
9309
OUTPUTS
0
1
2
3

S0	S1	FUNCTION
L	L	SUBTRACT
H	L	ADD
L	H	EX OR
H	H	AND

S2	S3	OPERATION
L	L	LOAD NEW DATA
H	L	NO SHIFT
L	H	SHIFT LEFT
H	H	SHIFT RIGHT

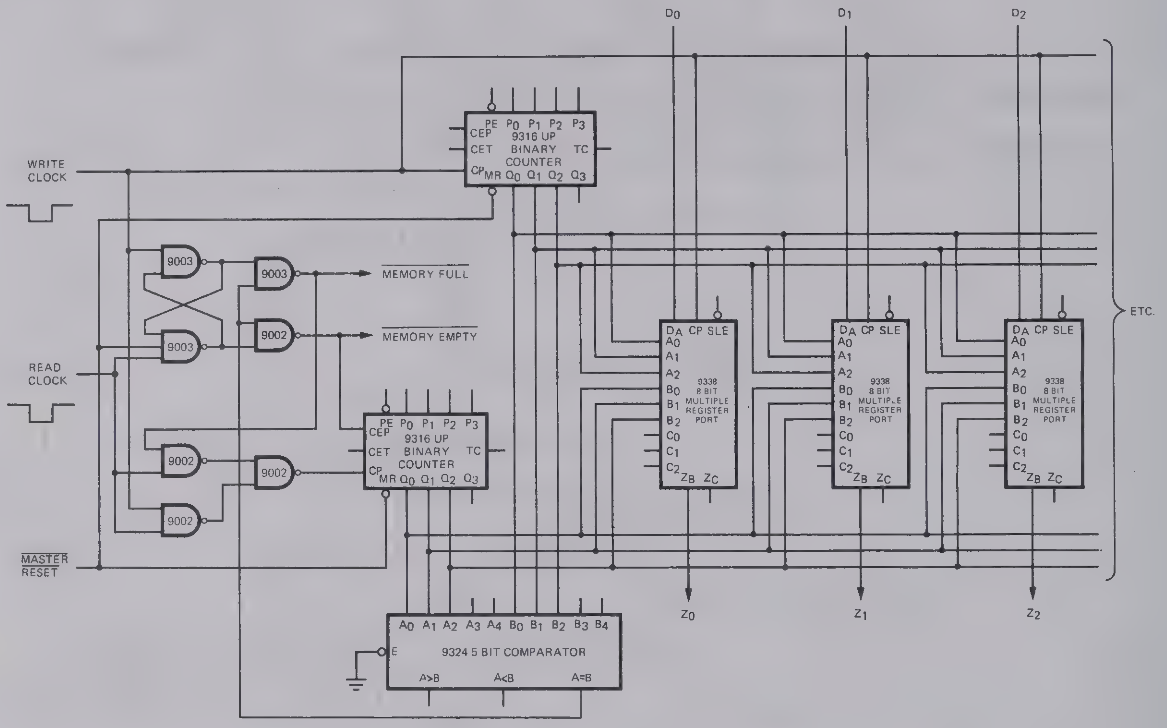
The basic application of the 9338 is as a 1-bit slice of eight registers/accumulators for use with an arithmetic logic unit. A 4-bit section of such a system is shown. This is easily expandable to a larger word length.

The 4-bit section uses four 9338 8-bit multiple port registers and two 9309 dual 4-input multiplexers with a 9340 4-bit arithmetic unit. The system provides eight storage registers that can be used as accumulators, data registers or scratch pad memories. Because of the versatility of the 9340, this design results in a flexible arithmetic unit. One or two words can be written into the 9340 and operated on according to the two select lines. The output can be fed back to the registers into the same or a new location, and it may be shifted left or right by the multiplexers before being reloaded. New data also enters through the multiplexers. For increased speed, two words can be operated on by the 9340 at the same time new data is being loaded.

This arithmetic unit is capable of operating by itself without additional access to the main memory. It can execute short subroutines very rapidly by using the eight registers of the 9338. The shifting capability provided by the multiplexers permits multiplication and division. With additional gating, this section can be part of a BCD arithmetic unit instead of the hexadecimal (binary) mode described here.

9338 8-BIT MULTIPLE PORT SHIFT REGISTER APPLICATIONS

FIRST IN/FIRST OUT MEMORY


D0
D1
D2
WRITE CLOCK
READ CLOCK
MASTER RESET
9003
9003
9003
9002
9002
9002
9002
MEMORY FULL
MEMORY EMPTY
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
DA CP SLE
A0 A1 A2 B0 B1 B2 C0 C1 C2
9338 8 BIT MULTIPLE REGISTER PORT
ZB ZC
Z0
Z1
Z2
ETC.
A0 A1 A2 A3 A4 B0 B1 B2 B3 B4
E
9324 5 BIT COMPARATOR
A>B
A<B
A=B

The First In/First Out memory above is capable of storing up to eight words. Separate write and read clocks allow asynchronous data entry and retrieval. Two binary counters (9316s) separately keep track of the read address and the write address.

In normal operation, the write address counter advances following each write clock and the read address counter advances following each read clock. Assuming one to seven words are stored in the 9338s, the read and write addresses will differ. Should a read clock cause the last word to be read from the memory or a write clock cause an eighth word to be written into the memory, the two counters will contain the same value. A 9324 comparator detects this condition and enables either the gate labeled Memory Full or the gate labeled Memory Empty depending on the state of the latch (two cross-coupled 9003 gates) determined by the last write or read clock. If the memory is empty, the read clock is not permitted to advance the read address counter. If the memory is full, additional write clocks advance both counters and cause the oldest data to be lost.

A Low on the Master Reset line initializes the memory by clearing both counters and setting the latch to the Memory Empty state. Care should be exercised in using this circuit if the Low-to-High transitions of both clocks can occur simultaneously. The state of the latch would be undeterminable if this occurred. Needless to say, the scheme is easily expanded to more words and to as many bits per word as needed by adding more 9338s, more counter stages, and buffers as needed in the address lines.

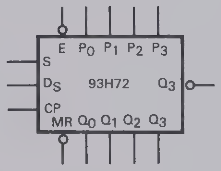
By using one 9366 up/down binary counter for both read and write address, a Last In/First Out memory can be designed. A write clock increments the counter following a write into the memory. A read clock causes the counter to decrement. The current address would be the most current word written into the 9338 bank. If two words are to be read without a write, then the latest and the next latest word would be read. The Terminal Count output from the 9366 can be used to prevent the counter from overflowing or underflowing.

93H72 HIGH SPEED 4-BIT SHIFT REGISTER WITH ENABLE

DESCRIPTION

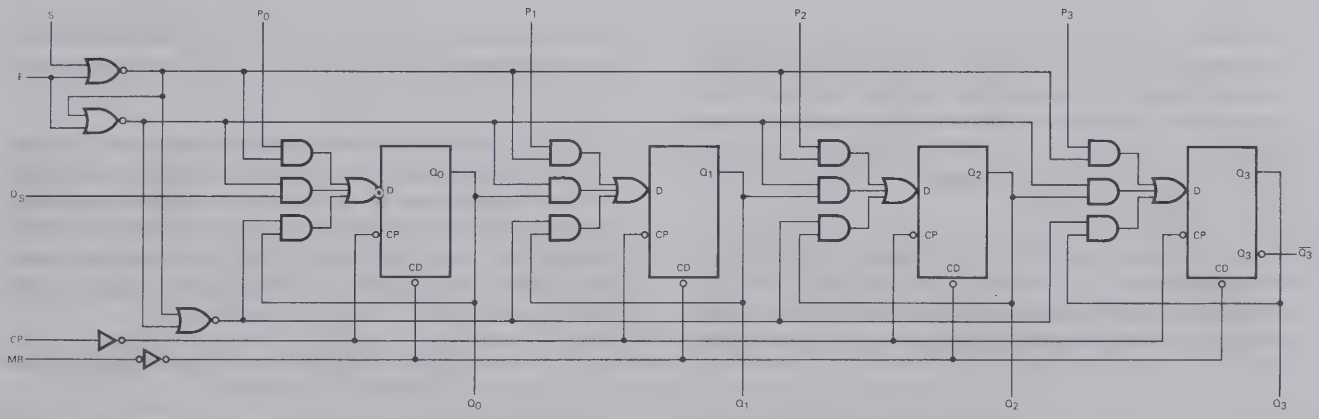
LEADS		LOADING
$\overline{E}$	Shift and Load Enable	2
S	Shift Select	1
P_0, P_1, P_2, P_3	Parallel Data Inputs	1
CP	Clock Input	2
$\overline{MR}$	Master Reset	1
Q_0 to Q_3, $\overline{Q}_3$	Parallel Outputs	10
D_S	Serial Data Input	1

CHARACTERISTICS	
TYPICAL SHIFT RATE	57 MHz
TYPICAL POWER DISSIPATION	400 mW
PACKAGE	16-lead DIP Flatpak



The 93H72 is a high speed 4-bit synchronous shift register with three modes of operation permitting serial and parallel data entry and data hold. The 93H72 can be used in any serial/parallel and input/output combination such as serial-serial, shift-left-and-right, parallel-to-serial and serial-parallel data transfers.

OPERATION



The 93H72 consists of four master/slave D flip-flops. The D input of each flip-flop can be logic switched among three sources; the parallel input, the output of the previous stage or the flip-flop's own output. The Shift Select and Enable inputs are decoded to provide source selection. The input for the first stage in the serial mode is a D type input. All flip-flops have asynchronous resets which are tied in common and buffered to provide the Master Reset capability.

The register is fully synchronous with information transfer to the outputs occurring when the clock goes from Low to High. The proper information for serial and parallel load need only appear on the serial or parallel inputs just prior to the Low-to-High clock transition. Synchronous operation allows any input/output combination, and in most cases a minimum of logic is required.

93H72 HIGH SPEED 4-BIT SHIFT REGISTER WITH ENABLE

MODE SELECTION

$\overline{MR}$	$\overline{E}$	S	D	Q_0	Q_1	Q_2	Q_3	
L	X	X	X	L	L	L	L	Asynchronous Clear
H	*h*	X	X	q_0	q_1	q_2	q_3	Hold
H	*l*	*h*	*h*	H	q_0	q_1	q_2	Shift right
H	*l*	*h*	*l*	L	q_0	q_1	q_2	Shift right
H	*l*	*l*	X	p_0	p_1	p_2	p_3	Parallel load

Lower case letters describe the state of the input (or output) one set-up time prior to the Low-to-High clock transition.

The 93H72 has three modes of operation. The active Low Enable ($\overline{E}$) input and Shift Select (S) input determine the mode of the register. A high on the $\overline{E}$ input overrides the Shift Select causing the register flip-flops to retain information. When the Enable is Low, the shift control operates the register in a shift right or parallel data entry mode.

When the Enable and Shift control inputs are both Low, the parallel inputs are selected and determine the next condition of the register synchronously with the clock. In this mode the element appears as four common clocked D flip-flops. With $\overline{E}$ Low and the Shift input High, the device acts as a 4-bit shift register with serial data entry through the D input.

The logic operation performed is determined by the logic levels on the D,P and S inputs one set-up time prior to the rising edge of the clock (non-ones catching). There is no other restriction on the activity of these inputs. The active Low Master Reset overrides all inputs and clears the register, forcing outputs Q_{0-3} Low and $\overline{Q}_3$ High.

The 93H72 4-bit shift register is similar to the 9300, but the J$\overline{K}$ shift inputs on the 9300 are replaced with a single D-type input on the 93H72. The 93H72 also has an active Low Enable with its associated "do nothing" mode. This active Low enable overrides all register operations except Reset and provides synchronous start and stop as well as shift and parallel load control. The active Low polarity of the Enable is compatible with other MSI devices such as decoders, allowing easy register selection in multiple register systems. The 93H72 can be used in all parallel/serial data transfer applications where the Low Enable with its associated hold mode is required.

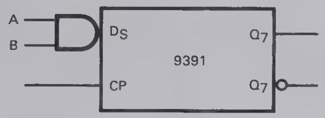
The 9300 (or 93L00, 93H00) should be used when serial J$\overline{K}$ input flexibility is required or when speed or reduced power consumption indicate the use of a low power or high speed 9300. An Enable can be incorporated on the 9300 version in shift applications by sacrificing parallel data entry and externally connecting the register output to the parallel inputs allowing data to recirculate as shown earlier. (There are similarities between the 8270/71 and the 93H72. However, the 8270/71 have overriding shift and the 93H72 has overriding disable. In addition, the 93H72 uses the more conventional clock polarity while the 8270/71 change on High-to-Low clock transition.)

9391/5491, 7491 8-BIT SHIFT REGISTER

DESCRIPTION AND OPERATION

LEADS		LOADING
A, B	Data Inputs	1 UL
CP	Clock Input	1 UL
Q_7, $\overline{Q}_7$	Data Output	10 UL

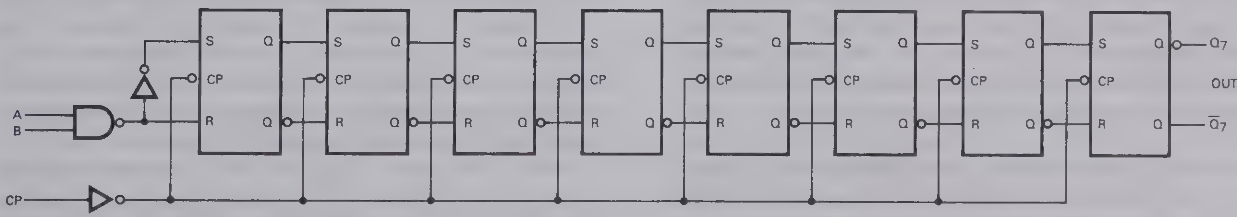
CHARACTERISTICS	
SHIFT FREQUENCY	18 MHz
POWER DISSIPATION	175 mW
PACKAGE	14-Lead DIP or Flatpak



t_n		$t_n + 8$
A	B	Q_7
L	L	L
L	H	L
H	L	L
H	H	H

t_n = bit time before clock

$t_n + 8$ = bit time after 8 clock pulses



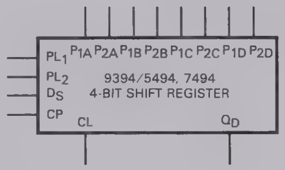
The 9391/5491, 7491 is a serial-in, serial-out 8-bit shift register for serial storage applications. The 9391 has eight R-S master/slave flip-flops, input gating and a clock driver. The register is capable of storing and transferring data at clock rates up to 18 MHz. The input to the first bit of the shift register is formed by 2-input AND gating. Drive for the internal common clock line is provided by an inverting clock driver. Each of the inputs (A, B, and CP) appears as only one TTL input load, and the output has 10 unit load drive capability. The AND gating facilitates data entry as shown in the truth table. This circuit shifts information to the output on the positive-going edge of an input clock pulse, making this shift register fully compatible with other MSI functions. Data must be present prior to the positive edge. The 9391 is an alternative to the 9328 when only eight bits of serial storage are required and a Master Reset is not needed.

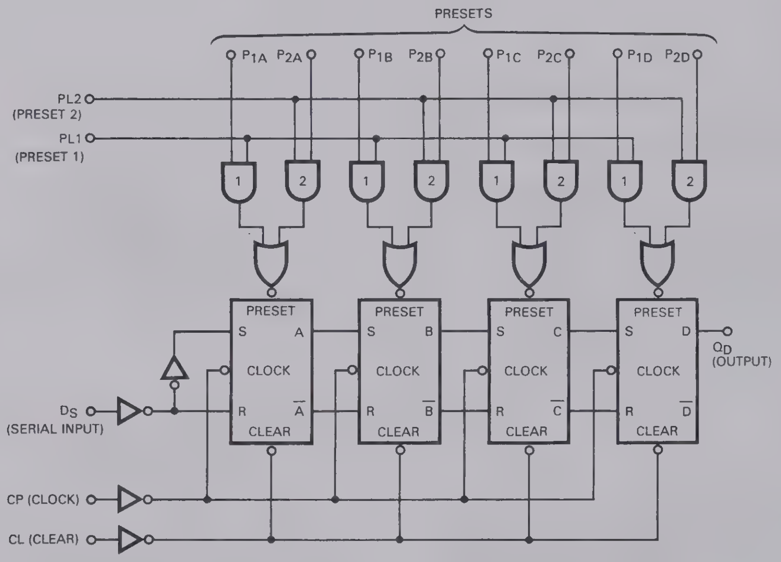
9394/5494, 7494 4-BIT SHIFT REGISTER

DESCRIPTION AND OPERATION

LEADS		LOADING
P_{1A} - P_{2D}	Preset Inputs	1 UL
PL_1	Preset 1 Input	4 UL
PL_2	Preset 2 Input	4 UL
D_S	Serial Data Inputs	1 UL
CP	Clock Input	1 UL
CL	Clear Input	1 UL
Q_D	Serial Data Output	10 UL

CHARACTERISTICS	
CLOCK FREQUENCY	15 MHz
PROPAGATION DELAY (CP to Q_D)	25 ns
POWER DISSIPATION	175 mW
PACKAGE	16-Lead DIP





The 9394 is a 4-bit shift register with serial and parallel (ones transfer) data entry. To facilitate parallel ones transfer from two sources, two parallel load inputs with associated parallel data inputs are provided. To accommodate these extra inputs only the output of the last stage is available. The asynchronous clear is active High; most MSI elements have active Low.

Four flip-flops are connected so that shifting is synchronous; they change state when the clock goes from Low to High. Data is accepted at the serial D input prior to this clock transition. The two parallel load inputs and parallel data inputs allow a non-synchronous ones transfer from two sources. The flip-flops can be set independently to the one state when the appropriate parallel input is activated. Preset inputs P_{1A} through P_{1D} are activated during the time the PL_1 is High and Preset inputs P_{1A} through P_{1D} are activated when PL_2 is High. If both sets of inputs are activated, a one on either input will set the flip-flops to a one. The register should not be clocked while the Preset Enables are activated. The active High Clear will override the clock and clear the register. The Preset Enable and active Preset input will override the clear if both are activated simultaneously. However, for predictable operation both signals should not be deactivated simultaneously.

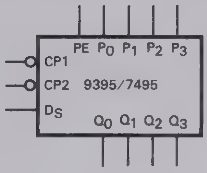
The 9394 can be used in parallel-to-serial conversions where four bits of parallel data from either of two sources must be converted to serial. However, the ones transfer form of parallel entry is cumbersome for data entry since zeros must be present in the register. This requires either asynchronously resetting the register or clearing it through the serial input. Once the register is initialized, parallel data can be preset into the register. Connecting the serial input Low as the information is shifted out shifts in zeros to clear the registers. The register must not be clocked while loading. Despite these minor inconveniences, the asynchronous parallel data entry is useful when an extra clock pulse for parallel load is impractical.

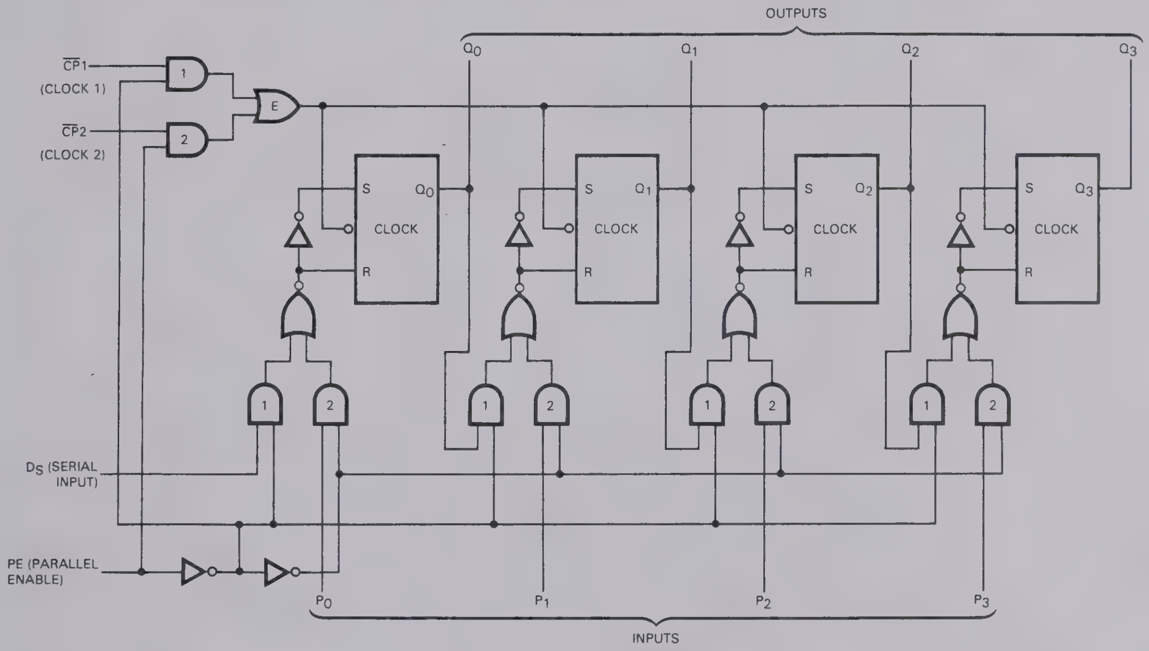
9395/5495, 7495 4-BIT SHIFT REGISTER

DESCRIPTION AND OPERATION

LEADS		LOADING
$\overline{CP}1$	Clock 1 Input	1 UL
$\overline{CP}2$	Clock 2 Input	1 UL
PE	Parallel Entry	2 UL
P_0, P_1, P_2, P_3	Parallel Data Inputs	1 UL
Q_0, Q_1, Q_2, Q_3	Parallel Data Outputs	10 UL
D_S	Serial Data Input	1 UL

CHARACTERISTICS	
CLOCK FREQUENCY	31 MHz
PROPAGATION DELAY (CP to Q)	25 ns
POWER DISSIPATION	250 mW
PACKAGE	14-Lead DIP





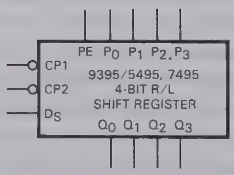
The 9395 is a 4-bit shift register which performs synchronous shift and synchronous parallel data transfers. Dual input clocks permit separate clock sources for the shift and parallel modes. Register clocking occurs on the negative (High-to-Low) transition of the clock's input; most MSI elements are Low-to-High. Several registers can be connected in series to form an n-bit right-shift or left-shift register. This register can also be used as a parallel-in, parallel-out storage register with gate (mode) control.

The 9395 is four master/slave flip-flops with AND/OR/INVERT gating to permit serial and parallel operation. When a Low level is applied to the PE control input, right-shift operation results. When a High level is applied, parallel data entry results. The PE control should only be changed when both clock inputs are Low. When a Low level is applied to the PE control input, the output of each flip-flop is coupled to the R-S inputs of the succeeding flip-flop and right-shift operation is performed by clocking at the clock 1 input. In this mode, serial data is entered at the serial input. Clock 2 and parallel inputs P_0 through P_3 are inhibited. When a High level is applied to the mode control input, the outputs are decoupled from the succeeding R-S inputs to prevent right-shift and the data entry is enabled through parallel inputs P_0 through P_3. This mode permits parallel loading of the register, or with external interconnection, shift-left operation. In this mode, shift-left can be accomplished by connecting the output of each flip-flop to the parallel input of the previous flip-flop (Q_3 to input P_2, etc.), and serial data is entered at input P_3.

Clocking for the shift register is accomplished through the AND/OR gate, (E) which permits separate clock sources to be used for the shift-right and parallel modes. If both modes can be clocked from the same source, a common clock input may be applied to clock 1 and clock 2. Information must be present at the R-S inputs of the master/slave flip-flops prior to clocking. Transfer of information to the outputs occurs when the clock input goes from High to Low.

9395/5495, 7495 4-BIT SHIFT REGISTER

TRUTH TABLES



PE	D_S	Q_0	Q_1	Q_2	Q_3	
l	*h*	H	q_0	q_1	q_2	Shift (on CP1)
l	*l*	L	q_0	q_1	q_2	Shift (on CP1)
h	X	p_0	p_1	p_2	p_3	Parallel load (on CP2)

Lower case letters describe the state of the input (or output) one set up time prior to the High-to-Low clock transition.
Shift is activated by clock CP1
Load is activated by clock CP2

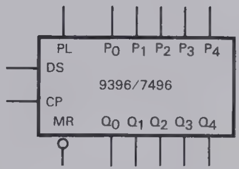
The 9395 is similar to the 9300 or 93H72 in that parallel and serial data operation is synchronous. However, unlike the 9300, the 9395 dual active Low clocks provide independent clocking for serial and parallel data entry. Also, the mode polarities are reversed with parallel data entry occurring when the parallel entry line (PE) is High on the 9395, Low on the 9300. Since the 9395 does have synchronous parallel and serial operation, it is as flexible in this respect as the 9300, 93L00 and 93H72, and it can be used in most of the applications shown for the 9300. However, mode change restrictions, the lack of Master Reset and $J\overline{K}$ inputs and the non-standard clock polarities (outputs change on a High-to-Low transition) must be acceptable. In some cases the independent clock control for serial and parallel operation can be used to advantage.

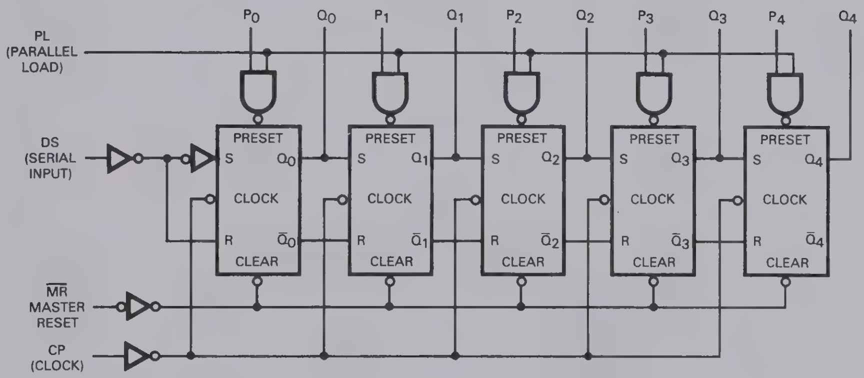
9396/5496, 7496 5-BIT SHIFT REGISTER

DESCRIPTION AND OPERATION

LEADS		LOADING
PL	Parallel Load Input	5 UL
P_0, P_1, P_2, P_3, P_4	Parallel Data Inputs	1UL
D_S	Serial Data Input	1 UL
$\overline{CP}$	Clock Input	1 UL
$\overline{MR}$	Master Reset Input	1 UL
Q_0, Q_1, Q_2, Q_3, Q_4	Parallel Data Outputs	10 UL

CHARACTERISTICS	
CLOCK FREQUENCY	15 MHz
PROPAGATION DELAY (CP to Q)	25 ns
POWER DISSIPATION	240 mW
PACKAGE	16-Lead DIP



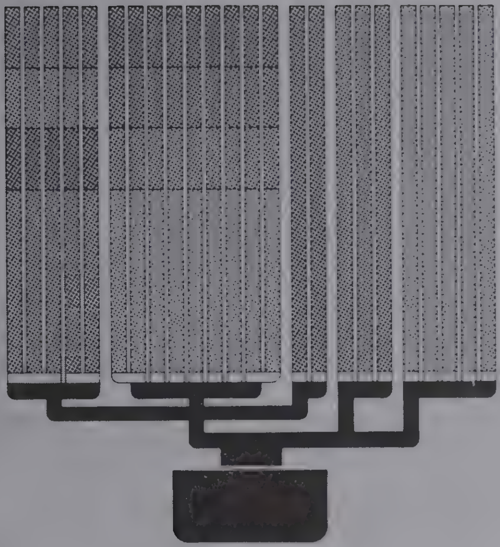


The 9396/5496, 7496 is a 5-bit shift register with both serial and parallel (ones transfer) data entry. Since the 9396 has the output of each stage available as well as a D-type serial input and ones transfer inputs on each stage, it can be used in 5-bit serial-to-parallel, serial-to-serial and some parallel-to-serial data operations.

The 9396 is five master/slave flip-flops connected to perform right shift. The flip-flops change state on the rising edge of the clock with data accepted at the serial D input prior to this clock transition.

Each flip-flop has asynchronous set inputs allowing them to be independently set to a one. The set inputs are connected through preset enable gating to allow non-synchronous parallel ones transfer. The flip-flops are independently set to the one state when a High logic level is present on the parallel input and the preset enable is High. The clear can be used to load zeros into the register, since preset inputs cannot force the flip-flops to the zero state. The register should not be clocked during the ones transfer operation. Each flip-flop has clear inputs connected together to provide reset capability. An active Low overrides the clock and clears the register. The preset inputs override the clear (forcing the output High) if both are activated simultaneously. However, for predictable operation, both signals should not be deactivated simultaneously.

The 9396 is the only 5-bit shift register and can be used when five bits of information must be handled. In such applications, the 9396 is useful not only as a 5-bit shift register but also for parallel-to-serial conversion. However, because of the ones transfer type parallel entry, the register must be cleared, either through the asynchronous reset or serial entry of zeros.



Counters

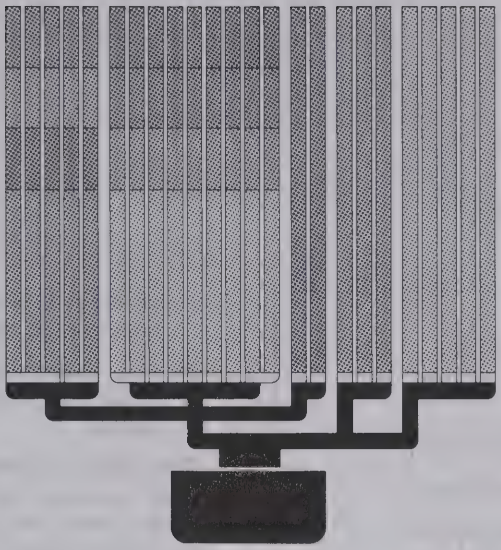
Overview
Asynchronous Dividers
Controlled Modulo Dividers
186 Different Divide Rates
Built-In Bounce Eliminator
Basic TV Sync Generators
2-Digit BCD Rate Multiplier
Fully Synchronous Presettable Counters
Programmable Dividers
Generating Complements
50% Duty Cycle Output Counters
Multistage Programmable Counters
Combination Counter/Serial Registers
Combination Counter/Decade Register
Cyclic D/A Conversion
Up/Down Counters
Light Controlled Up/Down Counting
Dead-Ended Counters
Synchronization and Coincident Pulse Prevention
Single Line Up/Down Control
Shift Register Counters
Twisted Ring Reversible Counters
Feedback Shift Register Counter
Divide-by-n Counters
Up/Down Counter with Multiplexed LED Display
Crystal Controlled Stopwatch

9
10
11
12
13
14
15

COUNTER SELECTION GUIDE

A — Asynchronous
S — Synchronous

Function	Device	Modulo	Parallel Load	Clock Transition	Max. Clock Rate, MHz (typ)	Clock to Q Output Delay ns (typ)	Power Dissipation, mW (typ)	Leads
Asynchronous	9390, 7490	2 x 5 (10)		↓	18	60	160	14
	9350	2 x 5 (10)		↓	18	60	160	14
	9391, 7492	2 x 6 (12)		↓	18	60	160	14
	9393, 7493	2 x 8 (16)		↓	18	75	160	14
	9356	2 x 8 (16)		↓	18	75	160	14
	93176, 74176	2 x 5 (10)	A	↓	50	45	150	14
	93177, 74177	2 x 8 (16)	A	↓	50	61	150	14
	93196, 74196	2 x 5 (10)	A	↓	70	38	240	14
	93197, 74197	2 x 8 (16)	A	↓	70	52	240	14
Semisynchronous	9305	2 x 5,6,7,8 (10,12,14,16)		↑	26	44	210	14
	93S05	2 x 5,6,7,8 (10,12,14,16)		↑	100	20	300	14
Synchronous	9310	10 (presettable)	S	↑	45	15	325	16
	93L10	10 (presettable)	S	↑	23	26	85	16
	93S10	10 (presettable)	S	↑	90	9	410	16
	9316	16 (presettable)	S	↑	45	15	325	16
	93L16	16 (presettable)	S	↑	23	26	85	16
	93S16	16 (presettable)	S	↑	90	9	410	16
Up-Down	9360,74192	10 (presettable)	A	↑	30	27	325	16
	9366, 74193	16 (presettable)	A	↑	30	27	325	16
	93190, 74190	10 (presettable)	A	↑	25	24	325	16
	93191, 74191	16 (presettable)	A	↑	25	24	325	16



COUNTERS

INTRODUCTION

Integrated circuits are used in a wide variety of counting applications in scientific instruments, industrial controls, computers and communications equipment as well as many other areas. Fairchild offers a large selection of MSI counter circuits, differing in complexity, functional versatility, speed, power and cost.

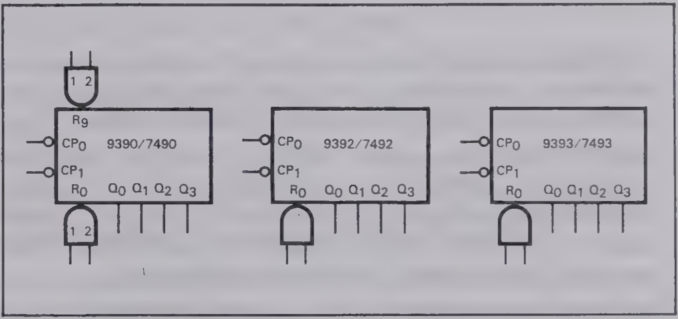
All of these MSI counter circuits consist of four master/slave flip-flops plus the necessary gating. With the exception of the 9392, they all normally count in a binary weighted code (binary or BCD) and they all have their four data bits brought out in parallel.

The counter selection guide includes the major parameters of all Fairchild MSI counters presently available. The counters are listed in order of complexity and cost. However, the lowest cost counter does not necessarily ensure the lowest cost system. The versatility of the more complex counters can often save external controls and buffer circuits, thus optimizing both system performance and cost.

ASYNCHRONOUS AND SEMISYNCHRONOUS COUNTERS WITHOUT PARALLEL LOAD

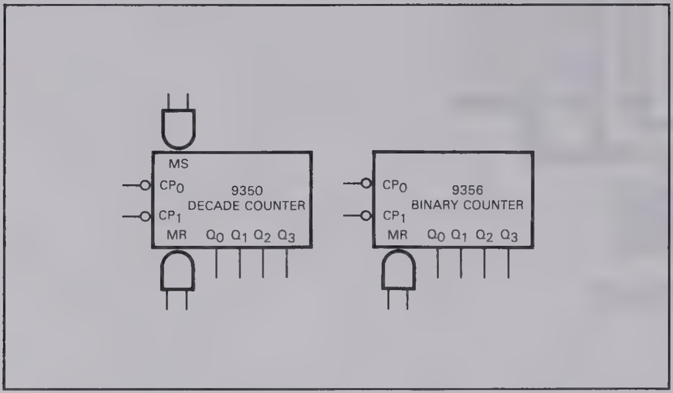
9390/7490, 9392/7492, 9393/7493

The simplest MSI counter is the 9390/7490. It is two separate counters, (a toggling flip-flop and a 3-stage modulo 5 counter) with separate Clock inputs and common Clear and Preset. For BCD (1248) counting, the output of the first stage is used as a clock input to the 3-stage counter. For biquinary counting, (50% duty cycle), the two counters are connected the opposite way with the modulo 5 counter feeding the modulo 2 counter. Asynchronous reset is provided through a 2-input AND gate (active High). Asynchronous preset to nine (preset first and last stage, clear the two center ones) is provided through another 2-input AND gate. The 9392/7492 is similar, but the second section divides by six. Therefore, the whole counter divides by 12. There is no Preset input. The 9393/7493 is the same, but its second section divides by eight, also with no Preset.



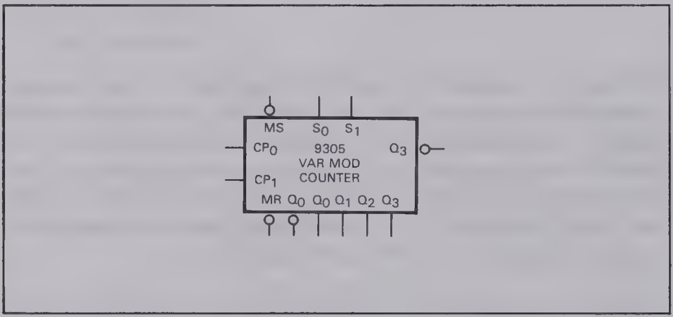
These counters have been available for several years. They are used in simple counting and frequency division applications when the improved features of more complex counters are not necessary, and when the application can tolerate the non-standard clock polarity (outputs change on the High-to-Low clock transition), the high clock loading (up to four unit loads) and the non-standard leads (V_{CC} lead 5, ground lead 10). Since these are ripple counters there is considerable delay between the clock and the change of the last flip-flop, particularly when several counters are cascaded. Decoding these counters thus generates glitches or spikes which can be minimized by using synchronous counters.

9350, 9356



The 9350 is the same circuit as the 9390/7490 and the 9356 is the same circuit as the 9393/7493 but with different connections. The power leads have been moved to the corners by rotating the chip inside the package. This is consistent with most TTL and MSI circuits simplifying PC board layout.

9305, 93S05



The 9305 Variable Modulo Counter is basically in the same category as the counters previously mentioned but is considerably more flexible, versatile and easier to use.

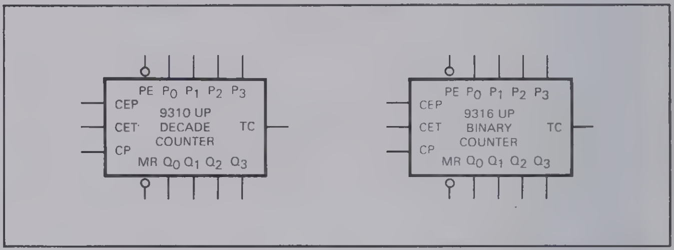
The 9305 has two separate counters, a toggling stage and a 3-stage counter with separate Clock inputs and common asynchronous Clear and Preset inputs. This device can substitute for the 9390/7490, 9392/7492, 9393/7493, 9350 and 9356. It also offers the following advantages.

- The 3-stage counter is synchronous, meaning less ripple delay and easier decoding.
- The 3-stage counter can be wired to divide by five, six, seven, or eight. Moreover, the Select inputs can be controlled by external gating to divide by any factor — even prime numbers. The applications section shows how two 9305s and one gate package can divide by any number up to 128 and by any even number up to 256.
- Both polarities of the first and last flip-flop outputs are brought out, simplifying systems design.
- The clock inputs are buffered, representing only one unit load. The polarity on the clock input is consistent with all other MSI sequential circuits (outputs change on the Low-to-High clock transition).
- V_{CC} and ground are on corner leads, consistent with most TTL and MSI circuits and simplifying PC board layout.

The 9305 is useful for counting or frequency division by a fixed number. It is faster with less ripple delay than the 9390/7490 and 9350 family counters, making decoding somewhat easier.

SYNCHRONOUS COUNTERS WITH SYNCHRONOUS PARALLEL LOAD

9310, 9316, 93L10, 93L16, 93S10, 93S16



The 9310 (BCD) and 9316 (binary) counters offer synchronous counting and synchronous parallel load. Any change of any flip-flop is controlled by the Low-to-High transition of the incoming clock. When the $\overline{\text{PE}}$ (active Low Parallel Enable) input is High and the Count Enable inputs are High, the clock causes the counter to increment. When the $\overline{\text{PE}}$ input is Low, the clock causes the four flip-flops to assume the state of the respective parallel input. Maximum counting frequency is comparable to the 9305, but the delay between the clock edge and any output change is typically only 15 ns, which is much faster than in the asynchronous (ripple) counters described above. Moreover, this synchronous operation can be extended up to 11 packages (44 bits) without additional components by proper use of the decoded Terminal Count (TC) output and the Count Enable inputs.

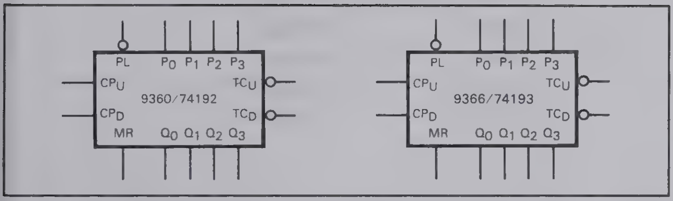
These counters are ideally suited for operation in synchronous systems. The synchronous parallel load and decoded terminal count make the design of arbitrarily presettable counters easy and free of any race conditions. The synchronous data inputs can be interconnected in such a way that the counter operates either as a shift register or counter controlled by the $\overline{\text{PE}}$ input. Data can be shifted out for a multiplexed display.

SYNCHRONOUS UP-DOWN COUNTERS

The counters already mentioned are unidirectional, counting up. Some applications require down counters, but often the design can be modified to use an up counter instead, e.g. by complementing the parallel input signals. In some cases, particularly in industrial applications, counters that count both up and down are needed.

COUNTERS

9360, 9366



The simplest bidirectional counters are the 9360/74192 (BCD decade) and 9366/74193 (binary) synchronous up/down counters with non-synchronous parallel data entry. These counters change state on the Low-to-High transition of either clock. Therefore, both clocks should not be Low simultaneously. Terminal Count Up (code 9 for BCD, code 15 for binary) and Terminal Count Down (code 0 for both counters) are decoded and brought out as active Low signals. This allows simple cascading of counters but results in a ripple delay of about 20 ns between counters.

The parallel data entry on these counters is non-synchronous, (independent of the clock), activated only by the active Low Parallel Load line. This can be an advantage since no clock pulse is required for loading the counter, but parallel load must be timed more carefully than in a synchronous scheme.

COUNTING WITH SHIFT REGISTERS

The MSI devices mentioned so far are designed as counters and are best suited for most counting applications. There are, however, cases where unconventional counting methods are more practical and economical.

Johnson (Moebius) Counters

The twisted-ring (Johnson or Moebius) shift register counter offers simple and glitch-free decoding with 2-input NAND gates. It uses n shift register stages for counting modulo 2n, therefore it is economical only for small values of n. The applications section shows the 9300 universal 4-bit shift register used as modulo 6 and 8 Johnson counters.

Linear Feedback Shift Register Counters

By applying a more sophisticated feedback, the 9300 can be used to count by any number up to 16, to count up and down and can easily be programmed, (e.g., as a programmable divider for a frequency synthesizer). Using the 9300 as a counter can offer two advantages. It reduces the number of parts required, and it can decrease component cost.

Serial Incrementers and Decrementers

In serial incrementer/decrementer systems the functions of data storage and arithmetic operation, normally combined in MSI counters, are completely separated. Data storage is performed in the least expensive way by shift registers (in extreme cases by long MOS shift registers). The arithmetic operation is performed by a carry flip-flop and by adders (four bits parallel) or one exclusive-OR (bit serial).

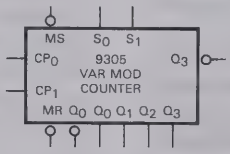
Serial incrementers are inherently slow, incrementing by one unit for a complete circular data shift. However, they offer several advantages.

- They can be easily gated to increment or decrement, resulting in up/down counters.
- Their modulo is easily controlled, simplifying the design of mixed modulo counters such as those for timers (seconds, minutes, hours, days).
- They combine naturally with display multiplexers resulting in economical display counters. The applications section includes a decimal 8-digit up/down counter with LED display and a digital stopwatch (milliseconds to hours) employing this technique.

9305 VARIABLE MODULO COUNTER

DESCRIPTION

LEADS		LOADING	
S_0, S_1	Select Inputs	1 UL	
CP_0	First Stage Clock (Active High Going Edge) Input	1 UL	
CP_1	Three Stage Clock (Active High Going Edge) Input	1 UL	
$\overline{MS}$	Master Set (Active Low) Input	1 UL	
$\overline{MR}$	Master Reset (Active Low) Input	1 UL	
		HIGH STATE	LOW STATE
Q_0	First Stage Output	16 UL	8 UL
$\overline{Q}_0$	Complementary First Stage Output	16 UL	8 UL
Q_1, Q_2, Q_3	Three Stage Counter Outputs	16 UL	8 UL
$\overline{Q}_3$	Complementary Last Stage Output	20 UL	10 UL

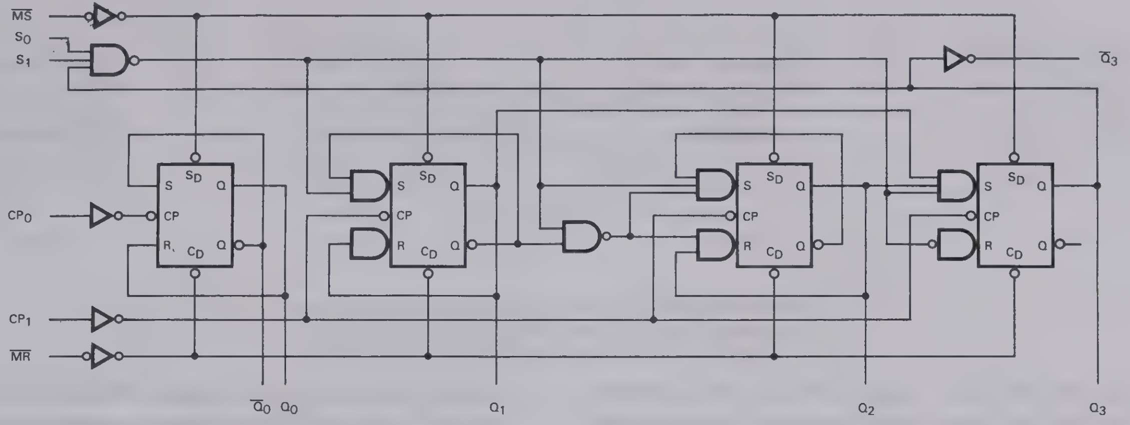


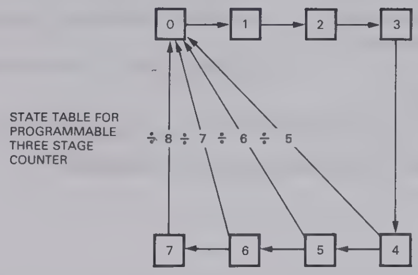
COUNTING MODES Without Additional Logic	FREQUENCY DIVISION MODES (With 50% Duty Cycle) Without Additonal Logic	ODD DIVISION MODES With Additonal Logic
Modulo 2	Modulo 8	Modulo 11
4	10	13
5	12	15
6	14	
7	16	
8		
10(1248 BCD)		
12		
14		
16		

The 9305 variable modulo counter is designed to serve in many counting and dividing applications in a digital system. The 9305 is an extremely versatile device, offering standard counter modes such as those tabulated without additional logic. In addition, the 9305 allows the design of difficult divide functions and divide/count functions economically both in terms of cost and logic required. The versatility of the 9305 is obvious from the many operational modes listed. The logic symbol, lead functions and loading rules are shown.

9305 VARIABLE MODULO COUNTER

OPERATION





The 9305 is four RS master/slave flip-flops. The four flip-flops are separated into two functional units, a single toggle stage (first flip-flop) and a 3-stage programmable synchronous binary counter (2nd, 3rd and 4th flip-flops). A clock buffer drives the first RS flip-flop toggle stage and a second clock buffer drives the three RS flip-flops in parallel to obtain 3-stage synchronous operation.

This sequence of events defines the operation of the master/ slave flip-flops relative to the clock input. When the Clock input (CP) is Low the slave is steady, but data can enter the master via the R and the S inputs. During the Low-to-High transition of CP, the data inputs (R and S) are inhibited so that a later change in the input data does not affect the master, and the information trapped in the master is transferred to the slave and appears at the outputs. When the transfer is completed, both the master and the slave are steady as long as the Clock input remains High, regardless of any other logic input except Master Reset ($\overline{MR}$) and Master Set ($\overline{MS}$). During the High-to-Low clock transition, the transfer path from master to slave is inhibited leaving the slave steady in its present state, and the data inputs (R and S) are enabled permitting new data to enter the master.

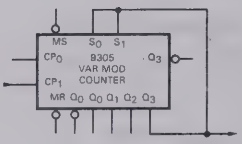
The first stage RS flip-flop is cross-coupled to provide a toggle stage. The RS flip-flops of the 3-stage counter are connected for synchronous binary counting. The synchronous AND reset with inputs S_0, S_1 and output Q_3 controls the modulo of the 3-stage counter by synchronously resetting all three flip-flops to zero when all inputs are High. One input of this gate is fed internally from the last counter output (Q_3) and prevents the counter from being synchronously reset before five binary states have occurred. Therefore, the count modulo depends on the programming connections made to these two inputs (S_0,S_1). For example, if the S_0 input is connected to the Q_2 output and the S_1 input is connected to the Q_1 output (or vice versa), the counter will count until Q_1, Q_2, and Q_3 are all High and then reset to zero synchronously with the clock to produce a modulo eight binary counter.

The complement output as well as the true output are available on the first toggle stage and last stage of the 3-stage counter. These complement outputs allow binary ripple clocking on the correct transition when the first toggle stage and 3-stage counter are connected together and/or are cascaded. The asynchronous Set and Reset are connected through noninverting buffers to the appropriate Set Direct and Clear Direct inputs of all four flip-flops.

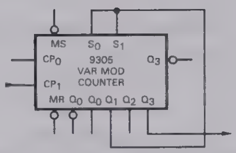
All inputs are buffered presenting only one unit load to the driving logic. Also, all the outputs have a high drive capability of at least eight unit loads. Both the low fan in and high fan out of the 9305 reduce the package count of a digital system (fewer external buffers are required) and improve performance (fewer buffers, less delay).

9305 VARIABLE MODULO COUNTER

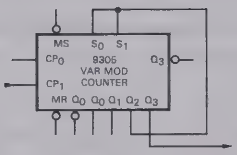
SYNCHRONOUS BINARY COUNT


MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3

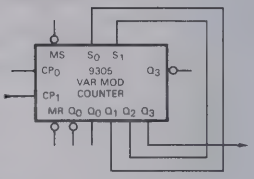
÷5


MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3

÷6


MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3

÷7


MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3

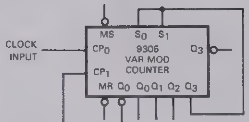
÷8

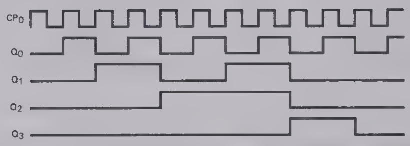
These connections provide the most convenient and most appropriate method of terminating unused Select inputs.

The 3-stage counter is programmed with external connections as shown providing synchronous binary count modulo five, six, seven, or eight. The programming of the three stages is independent of the toggle stage, and the basic configuration allows synchronous binary counting by the last three stages and modulo 2 operation by the first stage. Simultaneous frequency divisions of two, four and eight are available from the 3-stage counter when programmed to divide by eight.

STANDARD 4-BIT COUNTER CONFIGURATIONS

MODULO 10 COUNTER (BCD 1248 DECADE)

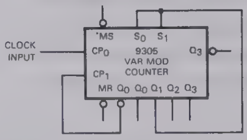

CLOCK INPUT
MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3

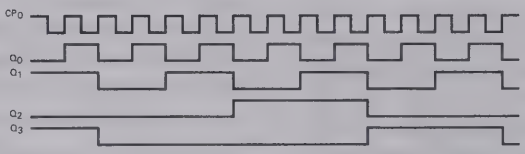


Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	0
0	0	0	1
1	0	0	1

MODULO 12 COUNTER

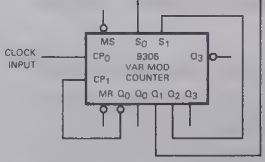

CLOCK INPUT
MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3

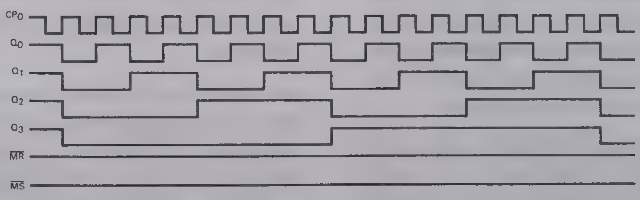


Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	0
0	0	0	1
1	0	0	1
0	1	0	1
1	1	0	1

MODULO 16 DIVIDER (4-BIT BINARY)


CLOCK INPUT
MS
S_0 S_1
CP_0
9305
VAR MOD
COUNTER
Q_3
CP_1
MR Q_0 Q_0 Q_1 Q_2 Q_3



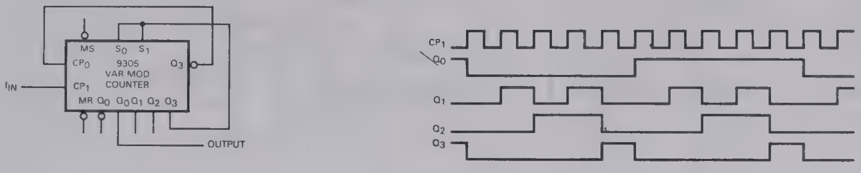
Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	0
0	0	0	1
1	0	0	1
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	1
1	1	1	1

9305 VARIABLE MODULO COUNTER

50% DUTY CYCLE OUTPUT DIVIDERS

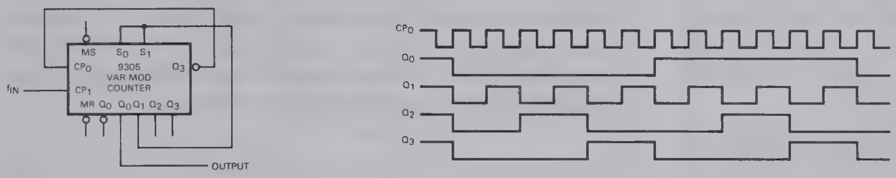
MODULO 10 DIVIDER, 50% DUTY CYCLE OUTPUT



Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
0	1	0	0
0	0	1	0
0	1	1	0
0	0	0	1
1	0	0	0
1	1	0	0
1	0	1	0
1	1	1	0
1	0	0	1

MODULO 12 DIVIDER, 50% DUTY CYCLE OUTPUT



Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
0	1	0	0
0	0	1	0
0	1	1	0
0	0	0	1
0	1	0	1
1	0	0	0
1	1	0	0
1	0	1	0
1	1	1	0
1	0	0	1
1	1	0	1

A 4-stage divider with 50% duty cycle output is produced by feeding the incoming clock to the 3-stage counter and clocking the single toggle stage with the $\overline{Q}_3$ output. Divider configurations of modulo 10 and 12 are illustrated. Modulo 16 division with 50% duty cycle is obtained from the $\overline{Q}_3$ output of the modulo 16 binary counter. In either the binary or 50% divison mode the modulo (10, 12, 14 or 16) is determined only by the external programming connections for the 3-stage counter. These 4-stage counters or dividers are not fully synchronous (semisynchronous) but have only one flip-flop ripple delay in any of these configurations. This flip-flop delay compares favorably with the more typical ripple delay of three in other counters, thus allowing higher operating speeds when counter states must be decoded. Also, 4-bit dividers with odd divide modulo can be formed with a few extra gates, illustrated later in 4-bit modulo dividers.

ASYNCHRONOUS MODE

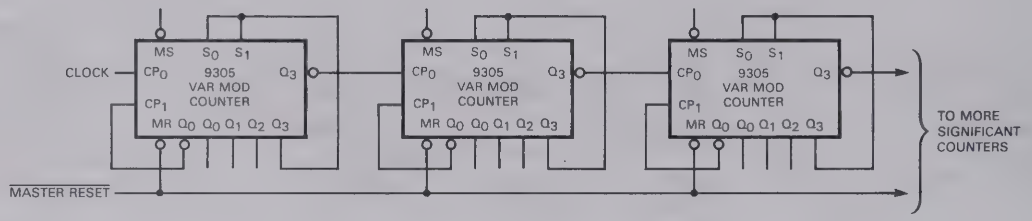
$\overline{MS}$	$\overline{MR}$	Q_0	$\overline{Q}_0$	Q_1	Q_2	Q_3	$\overline{Q}_3$
L	H	H	L	H	H	H	L
H	L	L	H	L	L	L	H
H	H	COUNT*					

*As determined by programming connections.

The Master Set and Reset asynchronously sets or resets all four stages when activated. The active Low Reset input, when Low, clears the counter overriding the clock and forcing the outputs Q_{0-3} Low and outputs $\overline{Q}_0$ and $\overline{Q}_3$ High. The active Low Set input, when Low, presets the counter overriding the clock and forcing the outputs Q_{0-3} High and outputs $\overline{Q}_0$ and $\overline{Q}_3$ Low. The Master Set provides a synchronous clear, since the first clock pulse following the asynchronous Master Set resets all stages. This action is independent of the modulo programmed.

9305 VARIABLE MODULO COUNTER APPLICATIONS

MULTISTAGE COUNTERS/DIVIDERS

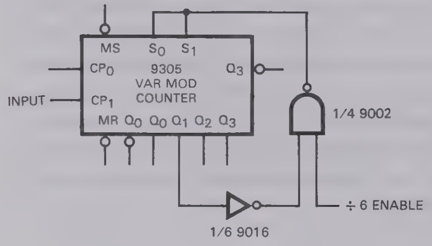

MS
S0
S1
CP0
9305
VAR MOD
COUNTER
CP1
Q3
MR Q0 Q0 Q1 Q2 Q3
CLOCK
MASTER RESET
TO MORE SIGNIFICANT COUNTERS

Variable Modulo 9305 counters can be connected together without additional logic to form ripple-type multistage counters or dividers. This is accomplished by connecting the active Low $\overline{Q}_3$ or $\overline{Q}_0$ output of the less significant counter to the Clock input of the following counter. The more significant counter is clocked when the less significant counter changes from its terminal count value to zero.

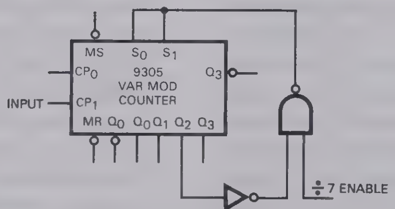
The ripple multistage counter formed with the 9305s has less ripple delay than conventional ripple counters, because it has only two ripple delay per four bits. The number of ripple delays for a typical 16-bit ripple binary counter would be 15, while a 16-bit counter composed of 9305s would have only seven ripple delays. In certain applications, the large ripple delay of conventional integrated 4-bit counters might dictate fully synchronous operation using more expensive counters.

COUNT MODULO CONTROL

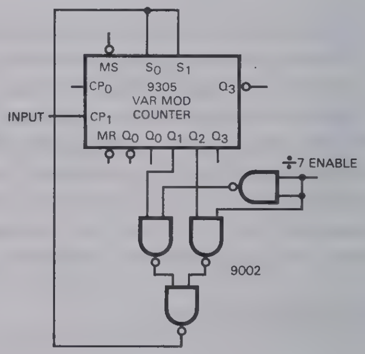
BETWEEN 5 AND 6


MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/6 9016
÷ 6 ENABLE

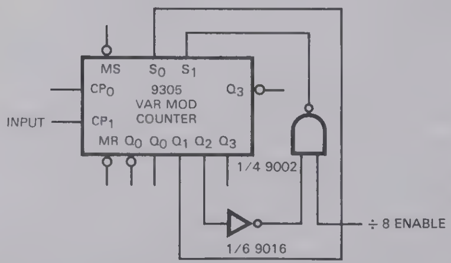
BETWEEN 5 AND 7


MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
÷7 ENABLE

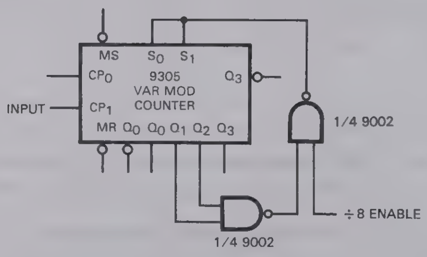
BETWEEN 6 AND 7


MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
÷7 ENABLE
9002

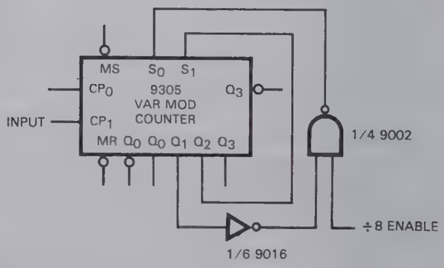
BETWEEN 6 AND 8


MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/6 9016
÷ 8 ENABLE

BETWEEN 5 AND 8


MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/4 9002
÷8 ENABLE

BETWEEN 7 AND 8

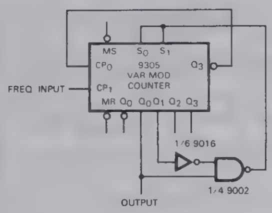

MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/6 9016
÷8 ENABLE

The count of the 9305 can be controlled with additional gating logic. This logic is inserted between the appropriate counter output(s) and the Select inputs, and controls the programmable 3-stage counter reset times. The amount of logic required depends on the number of different counts desired. Control between two counting levels is all that most applications require, although control between three or four distinct counting levels would be possible with additional logic. Such controllable modulo configurations can be used with the first stage or can be cascaded to produce counters and dividers with two or more count bases. These controllable configurations are the basis of odd modulo and multistage dividers.

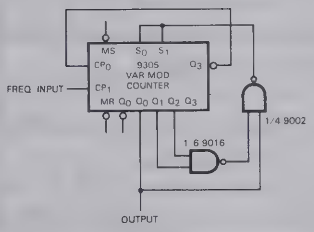
9305 VARIABLE MODULO COUNTER APPLICATIONS

ODD MODULO DIVIDERS

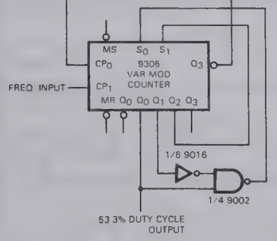
MODULO 11


MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
FREQ INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/6 9016
1/4 9002
OUTPUT

MODULO 13


MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
FREQ INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/6 9016
OUTPUT

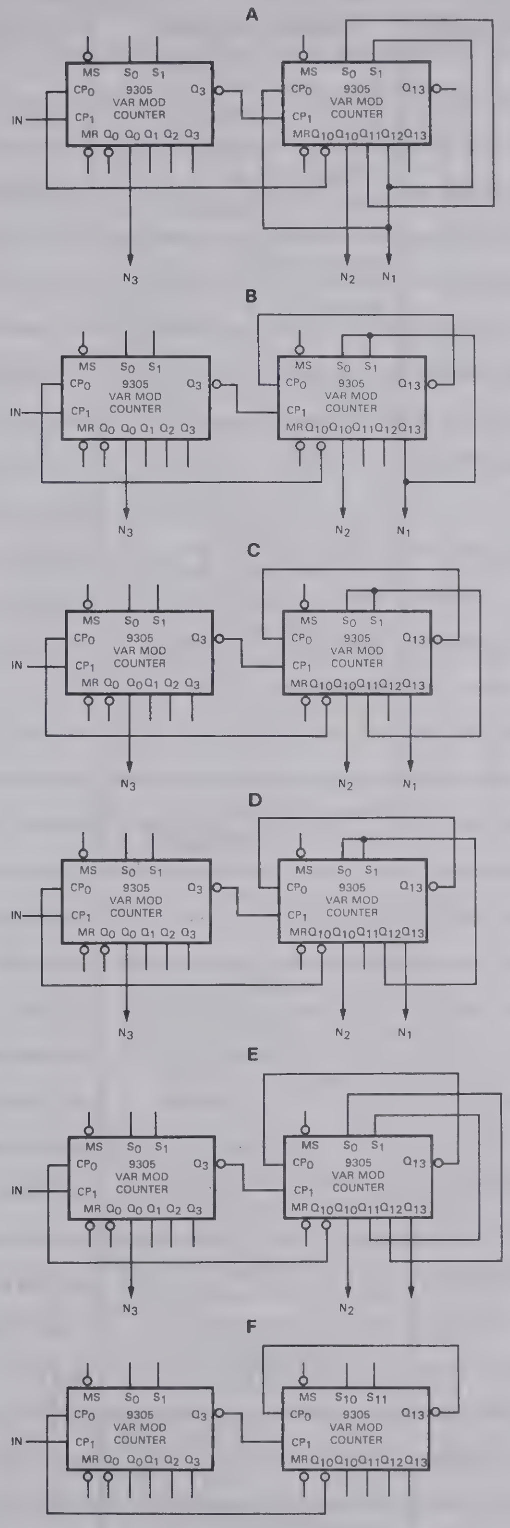
MODULO 15


MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
FREQ INPUT
CP1
MR Q0 Q0 Q1 Q2 Q3
1/6 9016
1/4 9002
53.3% DUTY CYCLE
OUTPUT

Odd modulo dividers use one of the several logic controllable counter configurations. The single toggle stage, 3-stage counter or other combinations of other 9305 counters are used to control the modulo and simultaneously divide the output of the controllable counter.

Illustrated are some 4-bit odd modulo dividers; modulo 11, modulo 13 and modulo 15. Each of these 4-bit dividers is easily implemented by adding two gates, or one gate and an inverter, to a single 9305. The first single stage in all of these dividers is clocked by the 3-stage counter and alternately switches the 3-stage counter's modulo between the upper and lower modulo The incoming clock is applied to the clock of the 3-stage counter, and the output is taken from the first stage. The output obtained for these 4-bit modulo counters is nearly 50% duty cycle. Full count speed is maintained since the control logic is not in a critical path. The first stage changes states (clocked) after the terminal state of the 3-stage counter, which is at least four states before the inputs to the select gates must be present and stable.

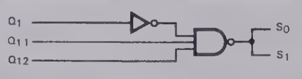
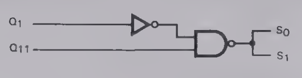
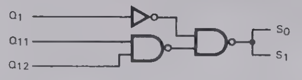
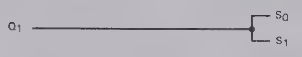
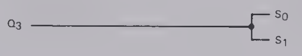
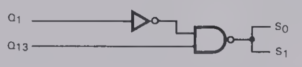
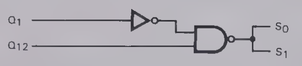
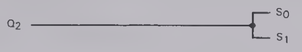
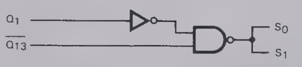
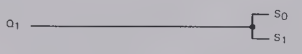
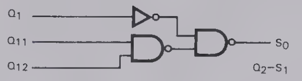
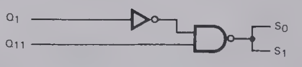
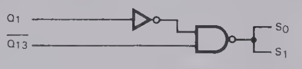
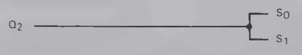
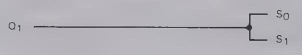
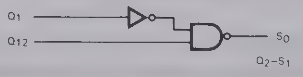
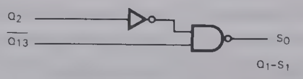
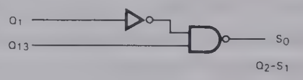
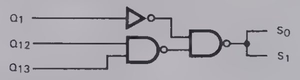
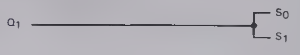
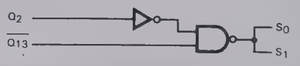
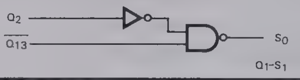
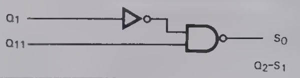
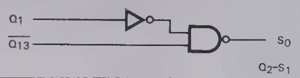
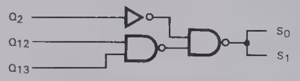
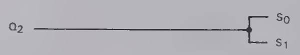
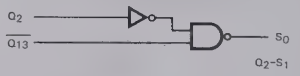
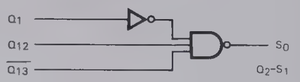
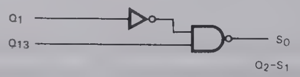
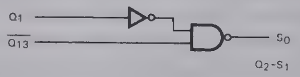
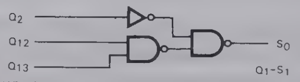
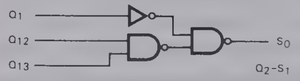
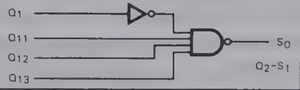
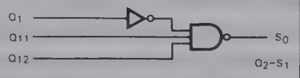
TWO 9305s AND ONE GATE PACKAGE PROVIDE 172 DIFFERENT DIVIDE RATES


A
B
C
D
E
F
MS
S0 S1
S10 S11
CP0
9305
VAR MOD
COUNTER
Q3
Q13
IN
CP1
MR Q0 Q0 Q1 Q2 Q3
MR Q10 Q10 Q11 Q12 Q13
N3
N2
N1

The 9305 variable modulo counter is particularly suited for low cost frequency dividers or counters that count by any fixed number, even primes. Division by any odd number from 21 to 127 and by any even number from 128 to 256 is possible with only two 9305 counters and one gate package. Six basic counter configurations are shown and the table lists the additional interconnections required for each division rate.

9305 VARIABLE MODULO COUNTER APPLICATIONS

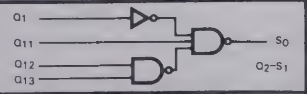
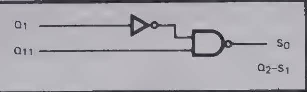
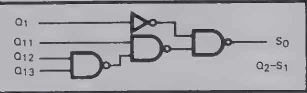
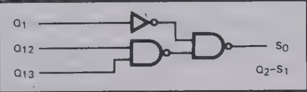
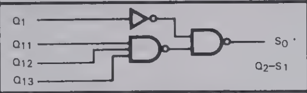
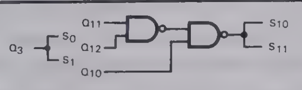
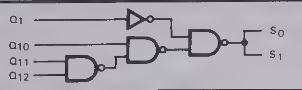
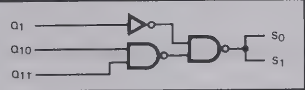
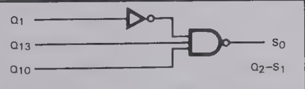
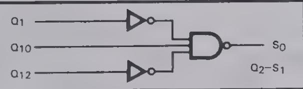
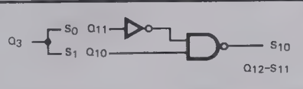
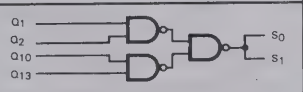
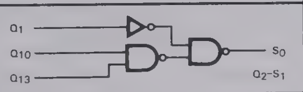
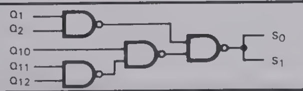
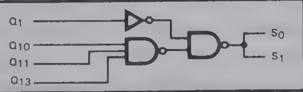
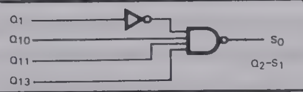
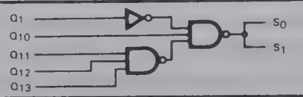
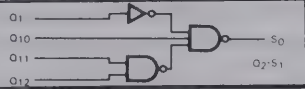
172 DIFFERENT DIVIDE RATE INTERCONNECTIONS

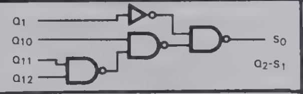
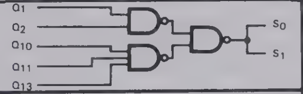
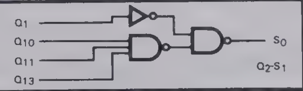
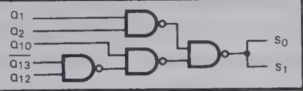
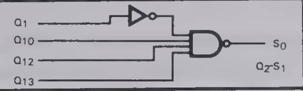
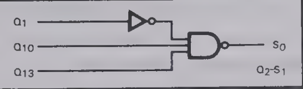
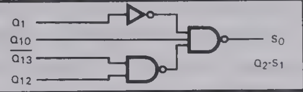
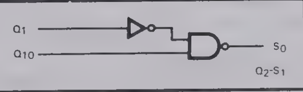
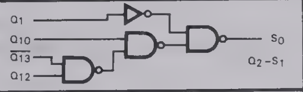
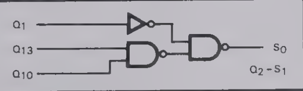
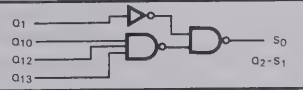
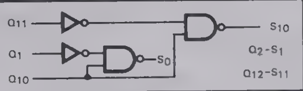
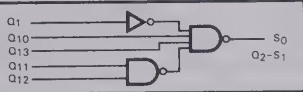
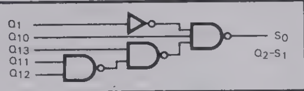
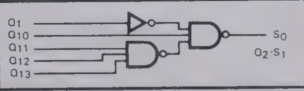
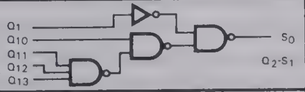
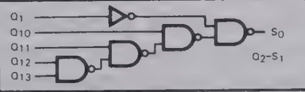
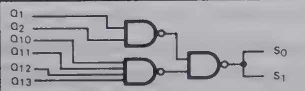
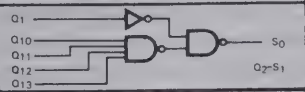
N_1	N_2	N_3	CC*	INTERCONNECTION	ALGORITHM
21	42	84	A	 Q1 Q11 Q12 S0 S1	21 = 3 x 5 + 1 x 6
22	44	88	A		22 = 2 x 5 + 2 x 6
23	46	92	A	 Q1 Q11 Q12 S0 S1	23 = 1 x 5 + 3 x 6
24	48	96	A		24 = 4 x 6
25	50	100	B		25 = 5 x 5
26	52	104	B		26 = 4 x 5 + 1 x 6
27	54	108	B		27 = 3 x 5 + 2 x 6
28	56	112	A		28 = 4 x 7
29	58	116	B	 Q1 $\overline{Q_{13}}$ S0 S1	29 = 1 x 5 + 4 x 6
30	60	120	B		30 = 5 x 6
31	62	124	A	 Q1 Q11 Q12 S0 Q2–S1	31 = 1 x 7 + 3 x 8
32	64	128	A	Q_2 - S_1 Q_1 - S_0	32 = 4 x 8
33	66	132	C		33 = 3 x 5 + 3 x 6
34	68	136	C	 Q1 $\overline{Q_{13}}$ S0 S1	34 = 2 x 5 + 4 x 6
35	70	140	B		35 = 5 x 7
36	72	144	C		36 = 6 x 6
37	74	148	B	 Q1 Q12 S0 Q2–S1	37 = 3 x 7 + 2 x 8
38	76	152	B	 Q2 $\overline{Q_{13}}$ S0 Q1–S1	38 = 1 x 6 + 4 x 8
39	78	156	B	 Q1 Q13 S0 Q2–S1	39 = 1 x 7 + 4 x 8
40	80	160	B	Q_1 - S_0 Q_2 - S_1	40 = 5 x 8
41	82	164	D	 Q1 Q12 Q13 S0 S1	41 = 1 x 5 + 6 x 6
42	84	168	D		42 = 7 x 6
43	86	172	D	 Q2 $\overline{Q_{13}}$ S0 S1	43 = 3 x 5 + 4 x 7
44	88	176	C	 Q2 $\overline{Q_{13}}$ S0 Q1–S1	44 = 2 x 6 + 4 x 8
45	90	180	C	 Q1 Q11 S0 Q2–S1	45 = 3 x 7 + 3 x 8
46	92	184	C	 Q1 $\overline{Q_{13}}$ S0 Q2–S1	46 = 2 x 7 + 4 x 8
47	94	188	D	 Q2 Q12 Q13 S0 S1	47 = 1 x 5 + 6 x 7
48	96	192	C	Q_1 - S_0 Q_2 - S_1	48 = 6 x 8
49	98	196	D		49 = 7 x 7
50	100	200	D	 Q2 $\overline{Q_{13}}$ S0 Q2–S1	50 = 3 x 6 + 4 x 8
51	102	204	D	 Q1 Q12 $\overline{Q_{13}}$ S0 Q2–S1	51 = 5 x 7 + 2 x 8
52	104	208	D	 Q1 Q13 S0 Q2–S1	52 = 4 x 7 + 3 x 8
53	106	212	D	 Q1 $\overline{Q_{13}}$ S0 Q2–S1	53 = 3 x 7 + 4 x 8
54	108	216	D	 Q2 Q12 Q13 S0 Q1–S1	54 = 1 x 6 + 6 x 8
55	110	220	D	 Q1 Q12 Q13 S0 Q2–S1	55 = 1 x 7 + 6 x 8
56	112	224	D	Q_1 - S_0 Q_2 - S_1	56 = 7 x 8
57	114	228	E	 Q1 Q11 Q12 Q13 S0 Q2–S1	57 = 7 x 7 + 1 x 8
58	116	232	E	 Q1 Q11 Q12 S0 Q2–S1	58 = 6 x 7 + 2 x 8

*CC = COUNTER CONFIGURATION

9305 VARIABLE MODULO COUNTER APPLICATIONS

172 DIFFERENT DIVIDE RATE INTERCONNECTIONS

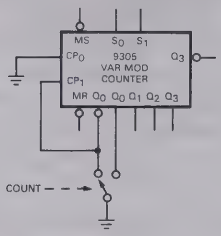
N_1	N_2	N_3	CC*	INTERCONNECTION	ALGORITHM
59	118	236	E	 Q1 Q11 Q12 Q13 S0 Q2-S1	59 = 5 x 7 + 3 x 8
60	120	240	E	 Q1 Q11 S0 Q2-S1	60 = 4 x 7 + 4 x 8
61	122	244	E	 Q1 Q11 Q12 Q13 S0 Q2-S1	61 = 3 x 7 + 5 x 8
62	124	248	E	 Q1 Q12 Q13 S0 Q2-S1	62 = 2 x 7 + 6 x 8
63	126	252	E	 Q1 Q11 Q12 Q13 S0 Q2-S1	63 = 1 x 7 + 7 x 8
64	128	256	E	Q_1 — S_0 Q_2 — S_1	64 = 8 x 8
	65	130	F	 Q11 S10 Q3 S0 Q12 S11 S1 Q10	65 = 13 x 5
	67	134	C	 Q1 S0 Q10 Q11 S1 Q12	67 = 5 x 5 + 7 x 6
	69	138	C	 Q1 S0 Q10 S1 Q11	69 = 3 x 5 + 9 x 6
	71	142	B	 Q1 Q13 S0 Q10 Q2-S1	71 = 9 x 7 + 1 x 8
	73	146	B	 Q1 Q10 S0 Q12 Q2-S1	73 = 7 x 7 + 3 x 8
	75	150	F	 S0 Q11 Q3 S1 Q10 S10 Q12-S11	75 = 15 x 5
	77	154	B	 Q1 Q2 S0 Q10 S1 Q13	77 = 1 x 5 + 9 x 8
	79	158	B	 Q1 Q10 S0 Q13 Q2-S1	79 = 1 x 7 + 9 x 8
	81	162	C	 Q1 Q2 S0 Q10 S1 Q11 Q12	81 = 5 x 5 + 7 x 8
	83	166	D	 Q1 S0 Q10 S1 Q11 Q13	83=1 x 5 +13 x 6
	85	170	C	 Q1 Q10 S0 Q11 Q2-S1 Q13	85=11 x 7 +1 x 8
	87	174	E	 Q1 S0 Q10 S1 Q11 Q12 Q13	87 = 9 x 5 + 7 x 6
	89	178	C	 Q1 Q10 S0 Q11 Q2-S1 Q12	89 = 7 x 7 + 5 x 8

N_1	N_2	N_3	CC*	INTERCONNECTION	ALGORITHM
	91	182	C	 Q1 Q10 S0 Q11 Q2-S1 Q12	91 = 5 x 7 + 7 x 8
	93	186	C	 Q1 Q2 S0 Q10 S1 Q11 Q13	93 = 1 x 5 +11 x 8
	95	190	C	 Q1 Q10 S0 Q11 Q2-S1 Q13	95 =1 x 7 +11 x 8
	97	194	D	 Q1 Q2 S0 Q10 Q13 S1 Q12	97 = 5 x 5 + 9 x 8
	99	198	D	 Q1 Q10 S0 Q12 Q2-S1 Q13	99 =13 x 7 +1 x 8
	101	202	D	 Q1 Q10 S0 Q13 Q2-S1	101 = 11 x 7 + 3 x 8
	103	206	D	 Q1 Q10 S0 Q13 Q2-S1 Q12	103 =9 x 7 + 5 x 8
	105	210	D	 Q1 Q10 S0 Q2-S1	105 =7 x 7 + 7 x 8
	107	214	D	 Q1 Q10 S0 Q13 Q2-S1 Q12	107 =5 x 7 + 9 x 8
	109	218	D	 Q1 Q13 S0 Q10 Q2-S1	109 = 3 x 7 + 11 x 8
	111	222	D	 Q1 Q10 S0 Q12 Q2-S1 Q13	111 = 1 x 7 + 13 x 8
	113	226	F	 Q11 S10 Q1 Q2-S1 S0 Q10 Q12-S11	113 =7 x 7 + 8 x 8
	115	230	E	 Q1 Q10 S0 Q13 Q2-S1 Q11 Q12	115 = 13 x 7 + 3 x 8
	117	234	E	 Q1 Q10 S0 Q13 Q2-S1 Q11 Q12	117 = 11 x 7 + 5 x 8
	119	238	E	 Q1 Q10 S0 Q11 Q2-S1 Q12 Q13	119 = 9 x 7 + 7 x 8
	121	242	E	 Q1 Q10 S0 Q11 Q12 Q2-S1 Q13	121 = 7 x 7 + 9 x 8
	123	246	E	 Q1 Q10 S0 Q11 Q2-S1 Q12 Q13	123 = 5 x 7 + 11 x 8
	125	250	E	 Q1 Q2 Q10 Q11 S0 Q12 S1 Q13	125 = 1 x 5 + 15 x 8
	127	254	E	 Q1 Q10 S0 Q11 Q12 Q2-S1 Q13	127 = 1 x 7 + 15 x 8

*CC = COUNTER CONFIGURATION

9305 VARIABLE MODULO COUNTER APPLICATIONS

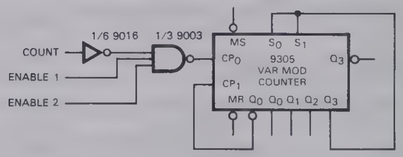
COUNTER WITH BUILT-IN BOUNCE ELIMINATOR


MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
CP1
MR Q0 Q0 Q1 Q2 Q3
COUNT

When IC counters are operated by push buttons or switches, the unavoidable contact bounce must be suppressed. Two cross-coupled NAND gates are commonly used to suppress bounce in a type C contact, but they are an additional component, increasing size and cost.

The first stage of a 9305 can be used as a debouncing latch. It is a master/slave flip-flop, but when the clock input CP_0 is Low, the master is isolated from the slave and Q_0 and $\overline{Q}_0$ can be used as latch outputs. This circuit accepts bouncing contacts and counts modulo 5, 6, 7, or 8, depending on the S_0 and S_1 connections to the $Q_{1,2,3}$ outputs. The steady state contact current is negligible (<2 mA), momentary current is ≈ 50 mA for < 20 ns.

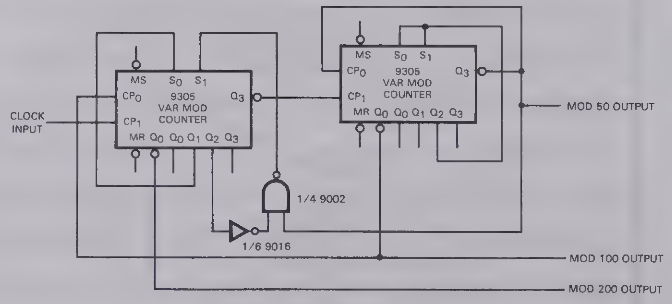
SINGLE/MULTIPLE COUNT ENABLES


1/6 9016
1/3 9003
COUNT
ENABLE 1
ENABLE 2
MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
CP1
MR Q0 Q0 Q1 Q2 Q3

Single or multiple count Enables can be added to the 9305 with an inverter and a NAND gate. The 9305 will not count if any of the count Enables are Low. Changes in the Enables should only be made when the clock is High.

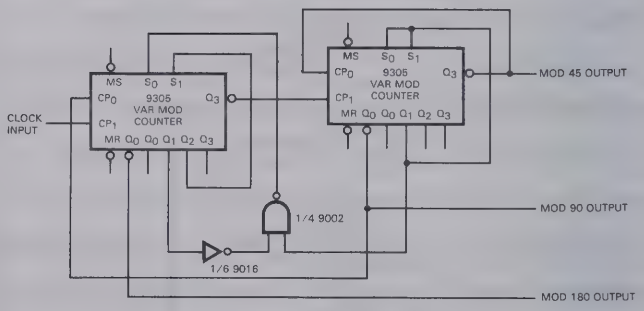
TWO TYPICAL COUNTERS

MODULO 50/100/200 DIVIDER


CLOCK
INPUT
MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/6 9016
MOD 50 OUTPUT
MOD 100 OUTPUT
MOD 200 OUTPUT

MODULO 50/100/200 DIVIDER

MODULO 45/90/180 DIVIDER


CLOCK
INPUT
MS
S0 S1
CP0
9305
VAR MOD
COUNTER
Q3
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/6 9016
MOD 45 OUTPUT
MOD 90 OUTPUT
MOD 180 OUTPUT

MODULO 45/90/180 DIVIDER

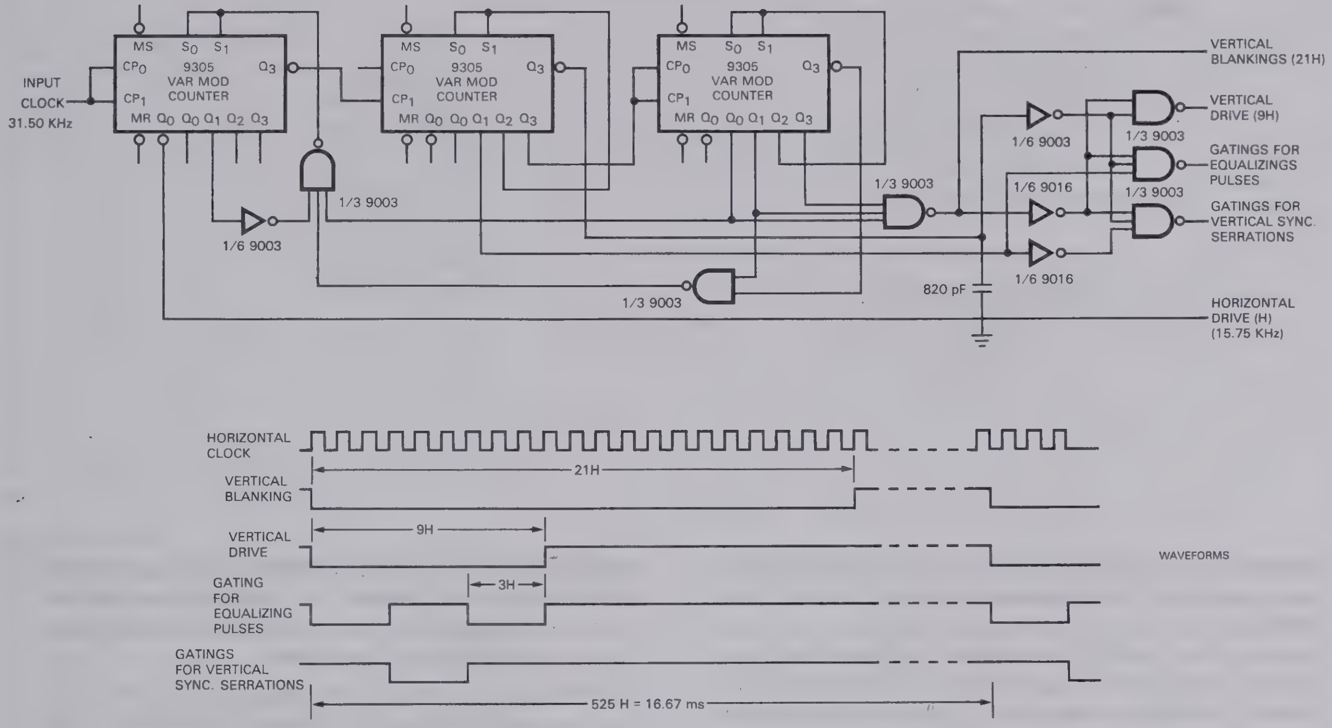
Two counter designs based on the information in the preceding table are shown. The 50/100/200 modulo divider uses a controllable modulo configuration of six and eight (first 9305) that clocks a modulo 7 binary counter. The control for the modulo 6 or 8 counter is taken from the complement of the last stage of the modulo 7 counter, which directs the controllable modulo counter to divide by six, three times, and by eight, four times. The modulo 50 output of the modulo 7 counter is divided by the first stage of each 9305 used, providing modulo 100 and modulo 200 division. The two toggle stages could operate independently if modulo 100 or modulo 200 division were not needed.

In the modulo 45/90/180 divider, the programmable 3-stage counter of the first 9305 divides by seven and eight. This 3-stage counter clocks the 3-stage programmable counter of the second 9305, which divides by six. The first stage output of the second 3-stage counter controls the seven/eight counter and directs it to divide by seven, three times, and by eight, three times. The modulo 45 output occurs at the $\overline{Q}_3$ output of the second 3-stage counter. As before, the first stages of each 9305 are used to divide the modulo 45 output further to obtain modulo 90 and modulo 180 division.

9305 VARIABLE MODULO COUNTER APPLICATIONS

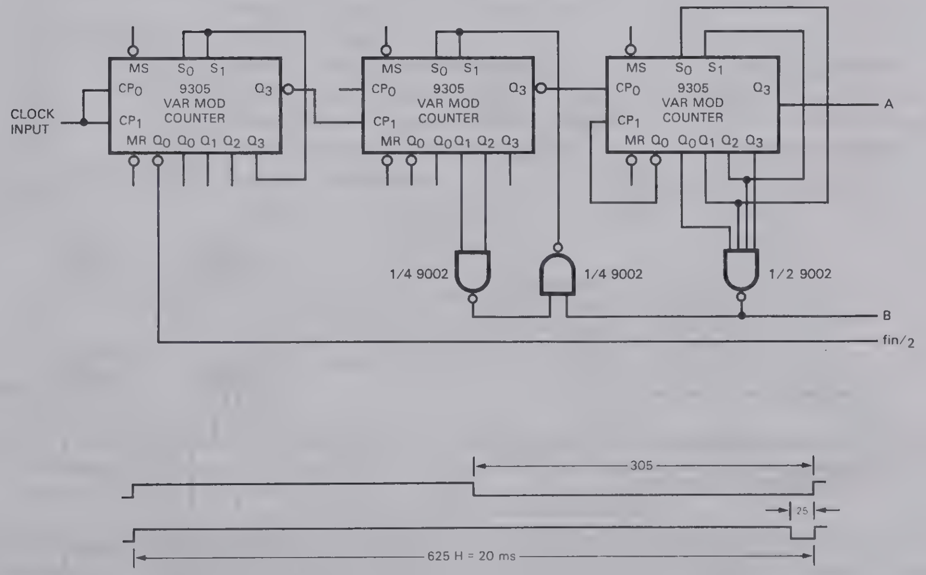
BASIC TV SYNC GENERATORS

MODULO 525



INPUT CLOCK 31.50 KHz
MS S0 S1
CP0 9305 VAR MOD COUNTER Q3
CP1
MR Q0 Q0 Q1 Q2 Q3
1/3 9003
1/6 9003
1/6 9016
820 pF
VERTICAL BLANKINGS (21H)
VERTICAL DRIVE (9H)
GATINGS FOR EQUALIZINGS PULSES
GATINGS FOR VERTICAL SYNC. SERRATIONS
HORIZONTAL DRIVE (H) (15.75 KHz)
HORIZONTAL CLOCK
VERTICAL BLANKING
VERTICAL DRIVE
GATING FOR EQUALIZING PULSES
GATINGS FOR VERTICAL SYNC. SERRATIONS
21H
9H
3H
525 H = 16.67 ms
WAVEFORMS

MODULO 625

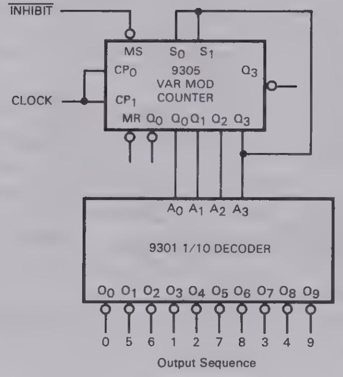


CLOCK INPUT
MS S0 S1
CP0 9305 VAR MOD COUNTER Q3
CP1
MR Q0 Q0 Q1 Q2 Q3
1/4 9002
1/4 9002
1/2 9002
A
B
fin/2
305
25
625 H = 20 ms

Two basic TV sync generators are illustrated above. The 525 divider produces waveforms basic to the American EIA standard and the waveforms of the modulo 625 divider are basic to the European standard. The waveforms of the modulo 525 divider and modulo 625 divider are the primary timing waveforms. By gating the waveforms shown with appropriate equalizing, vertical serration and blanking signals, standard sync waveforms can be generated.

9305 VARIABLE MODULO COUNTER APPLICATIONS

SYNCHRONOUS COUNTER/SEQUENCER


INHIBIT
MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
CLOCK
CP1
MR
Q0
Q0
Q1
Q2
Q3
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
0 5 6 1 2 7 8 3 4 9
Output Sequence

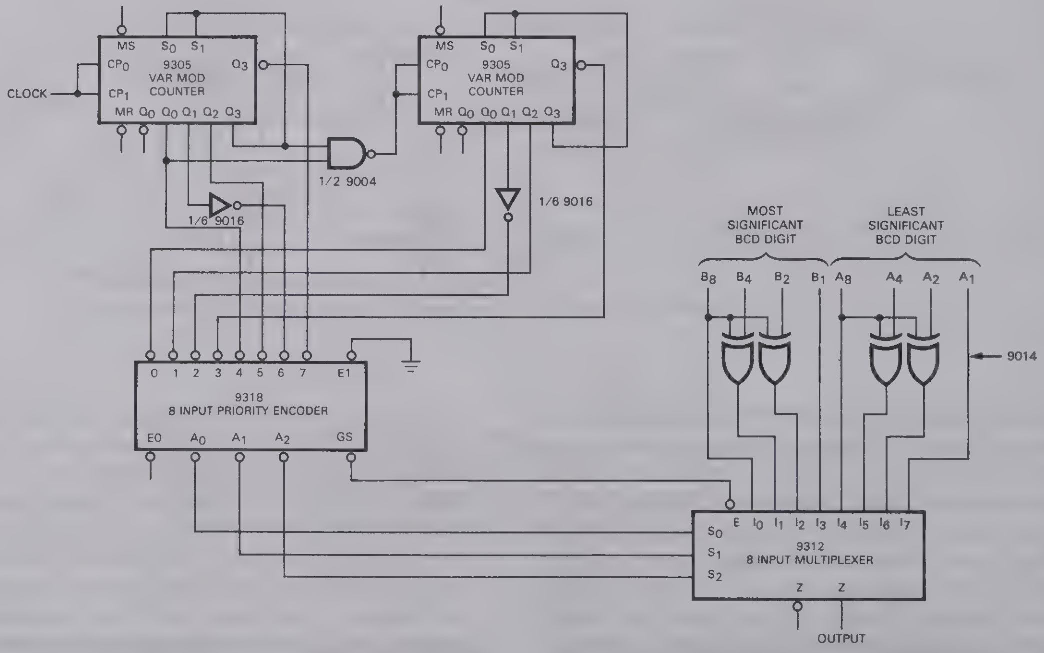
Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	1	0	0
0	0	1	0
1	1	1	0
0	0	0	1
1	0	0	0
0	1	0	0
1	0	1	0
0	1	1	0
1	0	0	1

Multistage semisynchronous or synchronous dividers are possible by simultaneously clocking two or more 9305 counters that are relative prime (have no common factor). This 1-of-10 sequencer uses the 9305 as a modulo 10 counter for generation of the decoder address. The 9305 is connected synchronously to minimize the delay between counter outputs and the corresponding transients on the decoder outputs. Synchronous operation is possible because the modulo of the first state (two) and the modulo of the 3-stage counter (five) are relative prime numbers. Ten unique states are produced by clocking the two counters simultaneously. The count sequence is also shown.

The Master Set inhibits the sequencer by setting all counter outputs ($Q_{0,1,2,3}$) to one, thus blanking the 9301 which does not recognize codes above nine. When the Master Set is deactivated, the counter sequences to zero synchronously with the next clock.

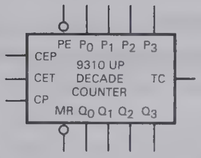
A 2-DIGIT BCD RATE MULTIPLIER

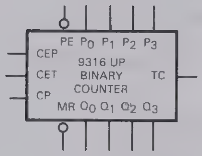

MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
CLOCK
CP1
MR Q0 Q0 Q1 Q2 Q3
MS
S0
S1
CP0
9305
VAR MOD
COUNTER
Q3
CP1
MR Q0 Q0 Q1 Q2 Q3
1/2 9004
1/6 9016
1/6 9016
MOST SIGNIFICANT BCD DIGIT
LEAST SIGNIFICANT BCD DIGIT
B8 B4 B2 B1 A8 A4 A2 A1
9014
0 1 2 3 4 5 6 7 E1
9318
8 INPUT PRIORITY ENCODER
EO A0 A1 A2 GS
S0
S1
S2
E I0 I1 I2 I3 I4 I5 I6 I7
9312
8 INPUT MULTIPLEXER
Z Z
OUTPUT

The 9305 counter can also act as the master counter in a BCD rate multiplying scheme. The 9305s are connected synchronously because this count sequence, as applied to the priority encoder, produces a code which gives an interlace sampling pattern for rate multiplying BCD 1248 numbers.

9310/93L10 AND 9316/93L16 COUNTERS

DESCRIPTION





LEADS		LOADING	
		9310/9316	93L10/93L16
$\overline{PE}$	Parallel Enable (Active Low) Input	2 UL	0.5 UL
P_0, P_1, P_2, P_3	Parallel Inputs	2/3 UL	0.167 UL
CEP	Count Enable Parallel Input	1 UL	0.25 UL
CET	Count Enable Trickle Input	2 UL	0.5 UL
CP	Clock (Active High) Going Edge Input	2 UL	0.5 UL
$\overline{MR}$	Master Reset (Active Low) Input	1 UL	0.25 UL
Q_0, Q_1, Q_2, Q_3	Parallel Outputs	8 UL	2 UL
TC	Terminal Count Output	10 UL	2 UL

$$\text{Count Enable} = CEP \bullet CET \bullet \overline{PE}$$

$$TC\ (9310) = CET \bullet Q_0 \bullet \overline{Q}_1 \bullet \overline{Q}_2 \bullet Q_3$$

$$TC\ (9316) = CET \bullet Q_0 \bullet Q_1 \bullet Q_2 \bullet Q_3$$

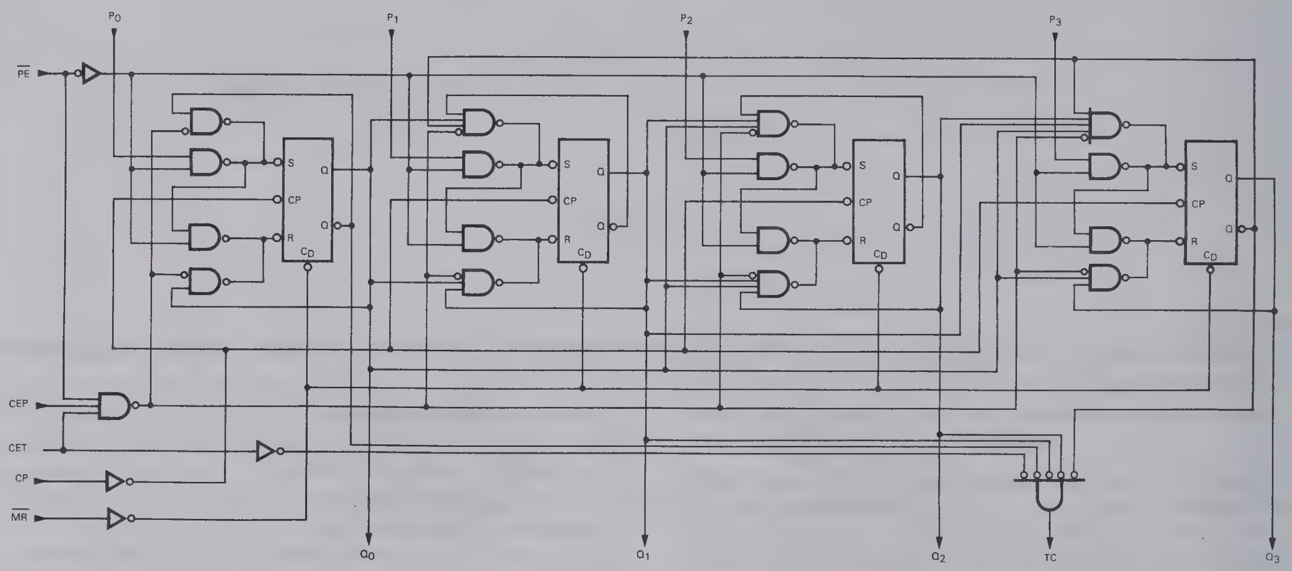
NOTE: $\overline{PE}$ and $\overline{MR}$ are active Low inputs, implying that they must be brought to a logic Low level in order to perform the function. Thus, data on $P_{0\text{-}3}$ is loaded into $Q_{0\text{-}3}$ by the rising edge of the clock when $\overline{PE}$ is Low. The counter is asynchronously reset when $\overline{MR}$ is Low.

The 9310/93L10 BCD decimal counters and the 9316/93L16 binary hexadecimal counters are multifunctional devices with the following features:

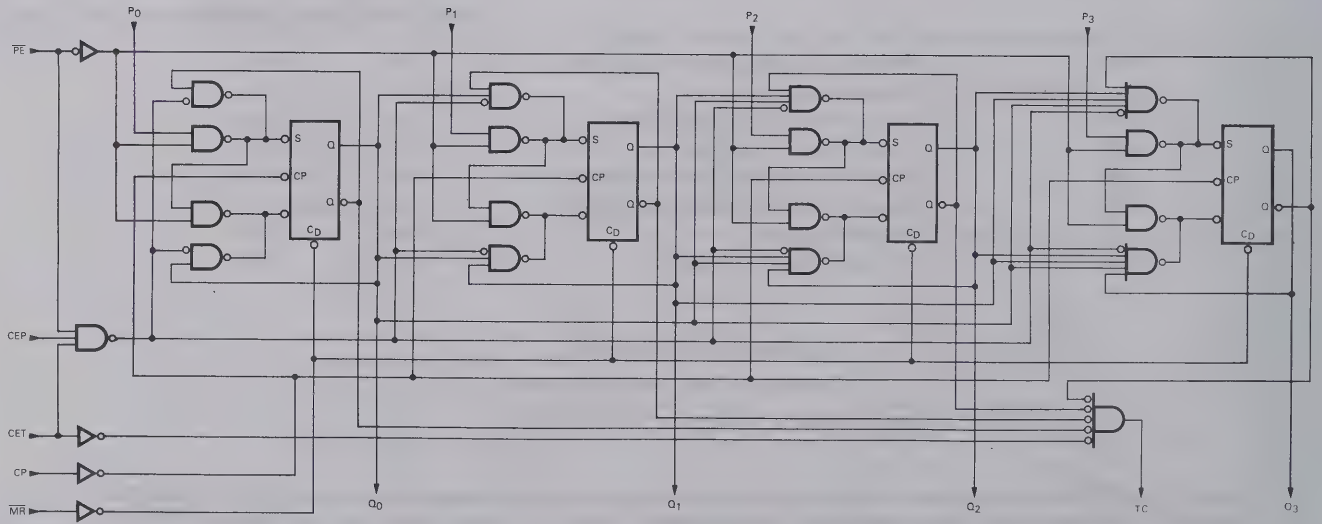
- Synchronous parallel loading
- Asynchronous Master Reset
- Assertion output from each stage
- Terminal count activated (High) at count 9 on the 9310/93L10 and at count 15 on the 9316/93L16
- Count Enable Parallel (CEP) input and Count Enable Trickle (CET) input permitting "enable while counting" in high speed multiple decade counting operations
- TTL integrated circuitry with active pullups to provide high speed with reasonable power consumption and excellent noise margins
- Input clamp diodes to ground to minimize the effects of line reflections

OPERATION

9310


P_0
P_1
P_2
P_3
$\overline{PE}$
S
Q
CP
R
C_D
CEP
CET
CP
$\overline{MR}$
Q_0
Q_1
Q_2
TC
Q_3

9316


P_0
P_1
P_2
P_3
$\overline{PE}$
S
Q
CP
C_D
CEP
CET
CP
$\overline{MR}$
Q_0
Q_1
Q_2
TC
Q_3

Logic diagrams for the 9310 and 9316 are shown. In each circuit, an inverting clock buffer drives the four clocked RS master/slave flip-flops in parallel to obtain synchronous operation. When the Clock input (CP) is Low, the slave is steady, but data can enter the master via the R and the S inputs. During the Low-to-High transition of the CP, data inputs R and S are inhibited so a later change in the input data does not affect the master, and the information trapped in the master is transferred to the slave and reflected at the outputs. When the transfer is completed, both the master and the slave are steady as long as the Clock input remains High, regardless of any other logic input except $\overline{MR}$. During the High-to-Low clock transition, the transfer path from master to slave is inhibited leaving the slave steady in its present state, and the data inputs R and S are *enabled* permitting new data to enter the master. This arrangement permits synchronous operation at high clock frequencies and simplifies control logic in most applications.

9310/93L10 AND 9316/93L16 COUNTERS

MODE SELECTION

$\overline{PE}$	CEP	CET	MODE
L	L	L	PRESET
L	L	H	PRESET
L	H	L	PRESET
L	H	H	PRESET
H	L	L	NO CHANGE
H	L	H	NO CHANGE
H	H	L	NO CHANGE
H	H	H	COUNT

Three control inputs — Parallel Enable ($\overline{PE}$), Count Enable Parallel (CEP) and Count Enable Trickle (CET) — select the mode of operation. There are three different modes: count, preset and no change.

Conventional operation of the 9310/9316, as shown above, requires that the mode control inputs ($\overline{PE}$, CEP, CET) are stable while the clock is Low. This is no constraint for a normal synchronous system where all signals are generated by the rising edge of the clock.

For some applications, the designer may want to change those inputs while the clock is Low. In this case the 9310/9316 will behave in a predictable manner. For example:

If $\overline{PE}$ goes High while the clock is Low, but Count Enable is not active during the remaining clock Low period (i.e., CEP or CET are Low), the subsequent Low-to-High clock transition will change $Q_0 \ldots Q_3$ to the $P_0 \ldots P_3$ data that existed at the set-up time before the rising edge of $\overline{PE}$.

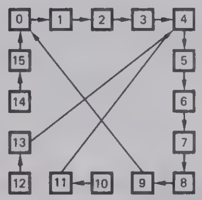
If Count Enable is active (i.e., CEP and CET are High) during some portion of the clock Low period, but $\overline{PE}$ is High (inactive) during the entire clock Low period, the subsequent Low-to-High clock transition will change $Q_0 \ldots Q_3$ to the next count value.

If $\overline{PE}$ goes High while the clock is Low, but Count Enable is active (CEP and CET are High) during some portion of the remaining clock Low period, the 9310/9316 will perform a mixture of counting and loading. On the Low-to-High clock transition, outputs $Q_0 \ldots Q_3$ will change as the count sequence or the loading requires. Only the outputs that would not change in the count sequence and are also reloaded with their present value stay constant.

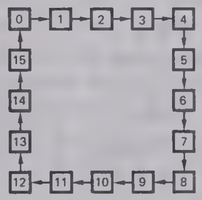
When Low, the asynchronous Master Reset overrides all other inputs resetting the four outputs Low.

COUNT SEQUENCE

9310



9316



Count Enable = CEP • CET • $\overline{PE}$

TC for 9310 = CET • Q_0 • $\overline{Q}_1$ • $\overline{Q}_2$ • Q_3

TC for 9316 = CET • Q_0 • Q_1 • Q_2 • Q_3

Preset = PE • CP+ (rising clock edge)

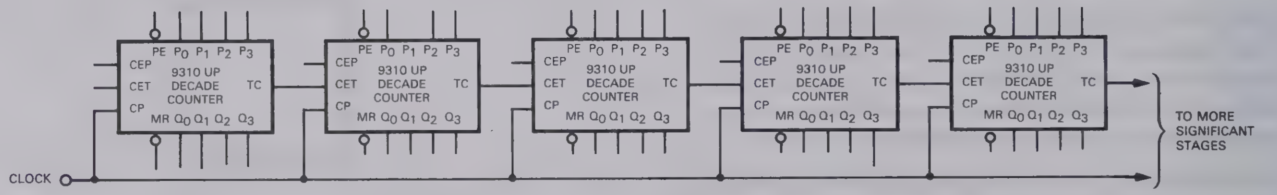
Reset = MR

The 9310 can be preset to any state, but will not count beyond 9. If preset to state 10, 11, 12, 13, 14, or 15, it returns to its normal sequence within two clock pulses.

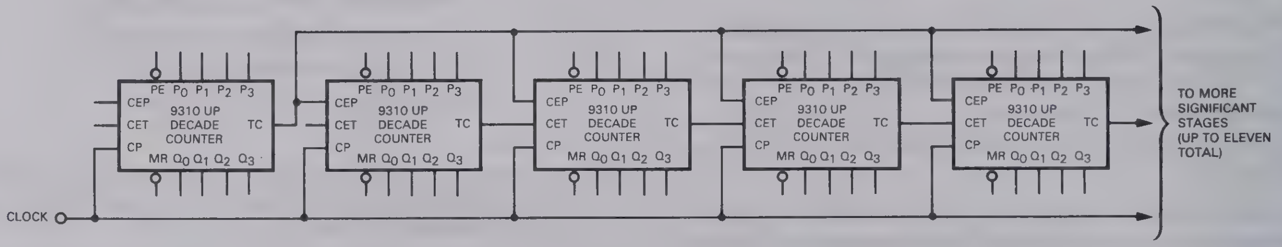
In each stage of the counter, the R and S inputs can be switched between two sources by the Parallel Enable ($\overline{PE}$) input. When the $\overline{PE}$ input is in the Low state, the R and S inputs of each of the four flip-flops are connected to form four independently clocked D flip-flops whose D inputs are the parallel inputs P_0, P_1, P_2 and P_3. This mode takes precedence over the two synchronous count enable controls, CEP and CET. The counters can be preset to any number between 0 and 15 by applying the binary equivalent of the number to the parallel inputs P_0, P_1, P_2 and P_3 and activating the $\overline{PE}$ input (PE Low). The counter is preset on the following Low-to-High clock transition. Counting is resumed when $\overline{PE}$, along with CEP and CET, is High and follows the sequence shown.

MULTISTAGE COUNTING

a. SLOW MULTISTAGE COUNTING SCHEME


PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
TO MORE SIGNIFICANT STAGES

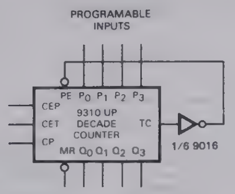
b. HIGH SPEED MULTISTAGE COUNTING SCHEME


PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
TO MORE SIGNIFICANT STAGES (UP TO ELEVEN TOTAL)

For multistage counting, all less significant stages must be at their terminal count before the next more significant counter is enabled. The 9310 and 9316 internally decode the terminal count condition which is ANDed with the CET input to generate the Terminal Count (TC) output. This arrangement allows series enabling by connecting the TC output (Enable signal) to the CET input of the following stage, *(a above).* This setup requires very few interconnections but has a drawback: the counter chain is fully synchronous, but since it takes time for the enable to ripple through the counter stages, maximum counting speed is reduced. This drawback can be overcome by proper use of the CEP and CET inputs. The CEP input of the 9310 or 9316 is internally ANDed with the CET input and connected to the R and S inputs of the individual flip-flops within the counter. This feature makes it possible to build a multistage counter *(b)* that can operate as fast as a single counter stage. The advantage of the "enable while counting" method is best seen by assuming all stages except the second and last are in their terminal condition. As the second stage advances to its terminal count, an Enable is allowed to trickle down to the last counter stage, but has the full cycle time of the first counter to reach it. When the TC of the first stage goes active (High), all CEP inputs are activated allowing all stages to count on the next clock. The fan out of TC (FO = 10) limits the configuration in *b* to 11 decades.

9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

PROGRAMMABLE DIVIDER


PROGRAMABLE INPUTS
PE
P0
P1
P2
P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR
Q0
Q1
Q2
Q3
1/6 9016

9310

DIVIDE RATIO	INPUT REQUIRED				
		P_0	P	P_2	P_3
2	8	L	L	L	H
3	7	H	H	H	L
4	6	L	H	H	L
5	5	H	L	H	L
6	4	L	L	H	L
7	3	H	H	L	L
8	2	L	H	L	L
9	1	H	L	L	L
10	0	L	L	L	L

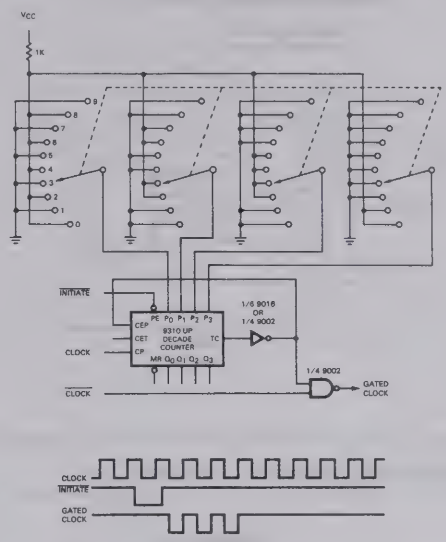
L = LOW = GROUND
H = HIGH = OPEN OR V_{CC}
(WITH 1 k LIMITING RESISTOR)

9316

DIVIDE RATIO	INPUT REQUIRED				
		P_0	P_1	P_2	P_3
2	14	L	H	H	H
3	13	H	L	H	H
4	12	L	L	H	H
5	11	H	H	L	H
6	10	L	H	L	H
7	9	H	L	L	H
8	8	L	L	L	H
9	7	H	H	H	L
10	6	L	H	H	L
11	5	H	L	H	L
12	4	L	L	H	L
13	3	H	H	L	L
14	2	L	H	L	L
15	1	H	L	L	L
16	0	L	L	L	L

The 9310 and 9316 can be converted into synchronous programmable counters with modulo 2 to 10 or 2 to 16 respectively, simply by adding a single inverter. The resulting counter actually simplifies certain applications. For example, most applications that seem to require a count either from zero to a predetermined number or from the number down to zero need not be implemented in this way. Instead, the operation can be performed with fewer gates by using a 9310 (or 9316) programmed with the complement of the number and allowed to count up until the terminal count is reached. Thus, if it is necessary to Enable a line for three counts, a 9310 may be used by entering 6 (0110) (9s complement of 3) on the parallel inputs and counting up.

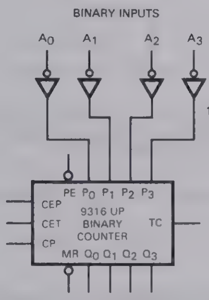
ROTARY SWITCH PROGRAMMING


VCC
1K
9
8
7
6
5
4
3
2
1
0
INITIATE
PE
P0
P1
P2
P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR
Q0
Q1
Q2
Q3
CLOCK
1/6 9016 OR 1/4 9002
1/4 9002
GATED CLOCK
CLOCK
CLOCK
INITIATE
GATED CLOCK

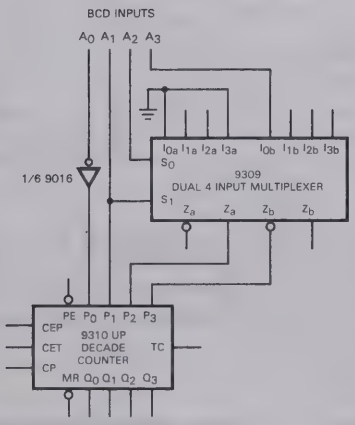
This 9310 is programmed by a 4-pole switch and is used to permit zero to nine pulses to pass through a gate. Various types of readily available thumbwheel switches function well with these devices. Because all 9300 series devices belong to the current sinking logic family, an open input appears to be High and must be grounded if a Low is desired. For this reason, nines complement switches providing closed contacts for High only will not function directly with these devices. A type providing shorts for Lows must be used. The proper type is sometimes called "nines complement plus complement" or "inverted nines complement" by thumbwheel switch manufacturers.

GENERATING COMPLEMENTS

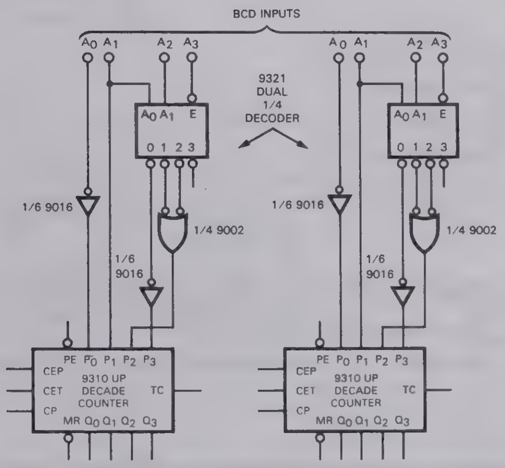
a.


BINARY INPUTS
A0
A1
A2
A3
1/6 9016 EACH
PE
P0
P1
P2
P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR
Q0
Q1
Q2
Q3

b.


BCD INPUTS
A0
A1
A2
A3
1/6 9016
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309 DUAL 4 INPUT MULTIPLEXER
Za
Za
Zb
Zb
PE
P0
P1
P2
P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR
Q0
Q1
Q2
Q3

c.

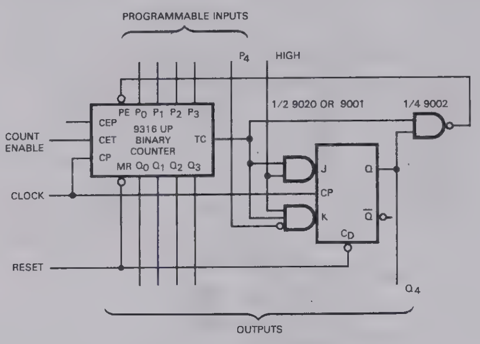

BCD INPUTS
A0
A1
A2
A3
A0
A1
A2
A3
9321 DUAL 1/4 DECODER
A0
A1
E
0 1 2 3
1/6 9016
1/4 9002
1/6 9016
PE
P0
P1
P2
P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR
Q0
Q1
Q2
Q3

The complement of a number may be generated with logic circuits. In a binary counter (9316), the negation outputs of the circuits supplying the number furnish the proper complement. If these outputs are not available, inverters will generate the complement as shown in *(a)* above. In a decimal counter (9310), the nines complement is generated by a 9309 dual 4-input multiplexer plus one inverter *(b)*, or by half of a 9321 dual 1-of-4 decoder plus a few gates, *(c)*.

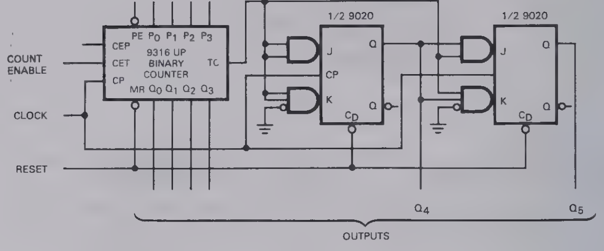
9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

COUNTER EXPANSION

PROGRAMMABLE MODULO 32

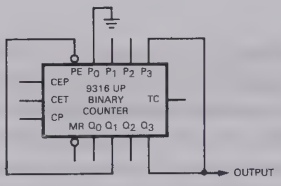

PROGRAMMABLE INPUTS
P4
HIGH
1/2 9020 OR 9001
1/4 9002
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
COUNT ENABLE
CLOCK
RESET
J
K
CP
Q
Q̄
CD
Q4
OUTPUTS

MODULO 64


1/2 9020
1/2 9020
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
COUNT ENABLE
CLOCK
RESET
J
K
CP
Q
CD
Q4
Q5
OUTPUTS

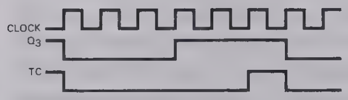
50% DUTY CYCLE OUTPUT COUNTERS

MODULO 6

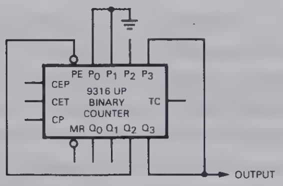

PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
OUTPUT

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
0	1	1	0
1	1	1	0
0	0	0	1
0	1	1	1
1	1	1	1

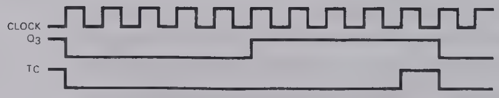


MODULO 10

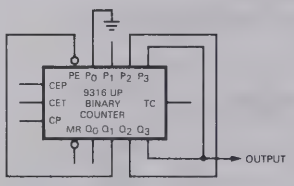

PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
OUTPUT

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	0
0	0	0	1
0	0	1	1
1	0	1	1
0	1	1	1
1	1	1	1

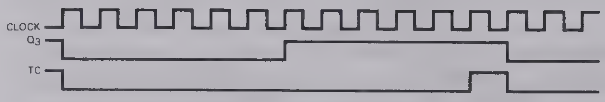


MODULO 12

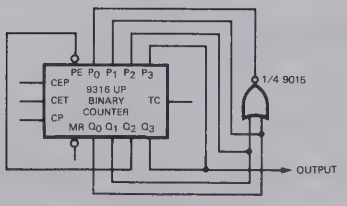

PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
OUTPUT

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
0	1	1	0
1	1	1	0
0	0	0	1
0	1	0	1
1	1	0	1
0	0	1	1
0	1	1	1
1	1	1	1

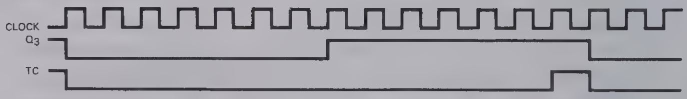


MODULO 14


1/4 9015
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
OUTPUT

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
0	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	0
0	0	0	1
1	0	0	1
0	1	0	1
0	0	1	1
1	0	1	1
0	1	1	1
1	1	1	1

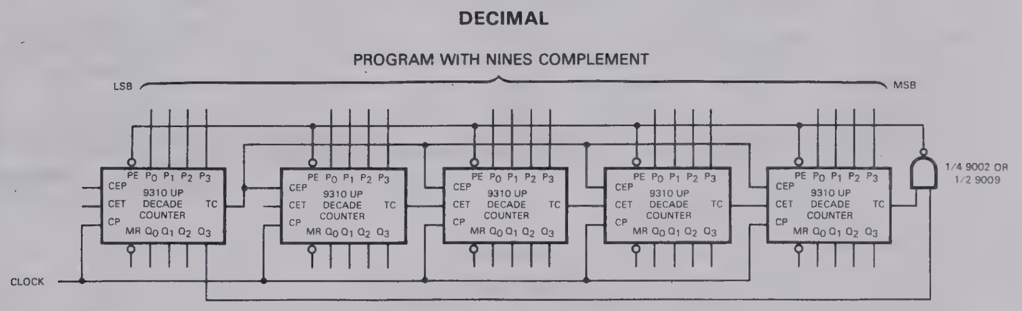


Four circuits which divide by 6, 10, 12 and 14 are shown. The Q_3 output provides a 50% duty cycle output. No additional gates are required except in the divide-by-14 circuit. In addition, all the count sequences start on 0000 and end on 1111, which means the Master Reset ($\overline{MR}$) input and the Terminal Count (TC) output still function properly.

Synchronously parallel loading the 9316 forces the counter to skip some of the states it would otherwise count through. In each circuit, either the Q_1 or Q_2 output is connected to the active Low Parallel Enable ($\overline{PE}$) input. Whenever this output is Low, the counter loads instead of counting on the next clock pulse.

9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

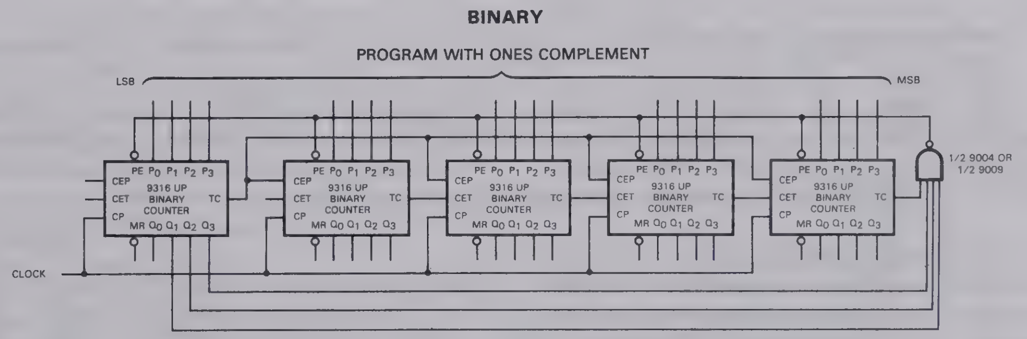
MULTISTAGE PROGRAMMABLE COUNTERS



DECIMAL

PROGRAM WITH NINES COMPLEMENT

LSB
MSB
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
1/4 9002 OR 1/2 9009
CLOCK



BINARY

PROGRAM WITH ONES COMPLEMENT

LSB
MSB
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
1/2 9004 OR 1/2 9009
CLOCK

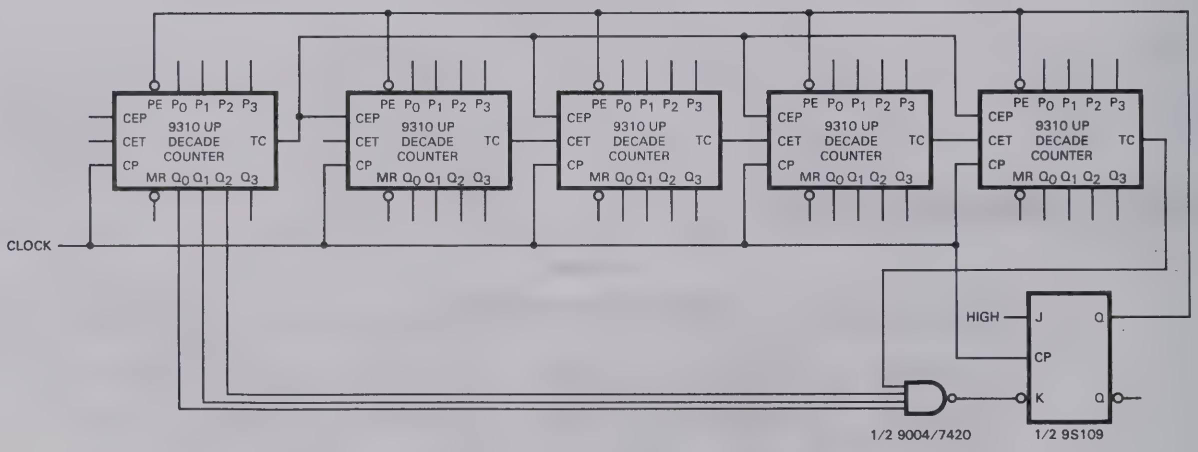
In the multistage programmable decimal and binary counters shown above the state prior to Terminal Count (TC-l) is decoded and activates the $\overline{PE}$ input. Therefore, the next clock pulse does not increment the counter to Terminal Count (all nines for decimal, all ones for binary) but rather loads the program value into the counter. The counters are programmed with the nines or ones complement of the count modulos, instead of the more complicated tens or twos complement used in the conventional approach. The maximum count frequency is limited by the delay in TC decoding and the setup time of the $\overline{PE}$ input. This can be improved with an additional flip-flop as shown next.

9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

MULTISTAGE PROGRAMMABLE COUNTERS

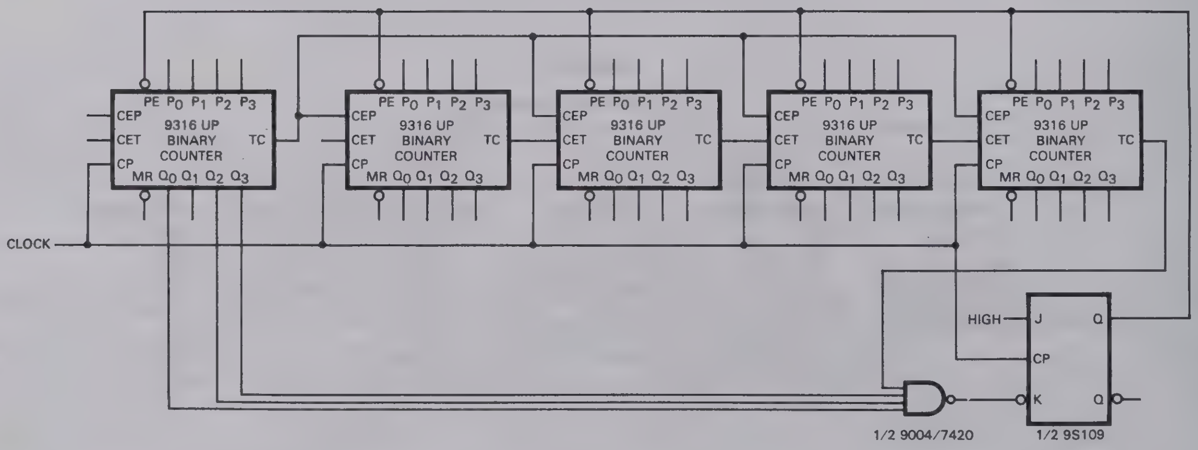
DECIMAL

PROGRAM WITH NINES COMPLEMENT


PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
HIGH
J
CP
K
Q
1/2 9004/7420
1/2 9S109

BINARY

PROGRAM WITH ONES COMPLEMENT


PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
HIGH
J
CP
K
Q
1/2 9004/7420
1/2 9S109

The maximum count frequency of a programmable counter can be improved by decoding the TC-2 state of the counter and synchronizing this state in a fast flip-flop such as the 9S109.

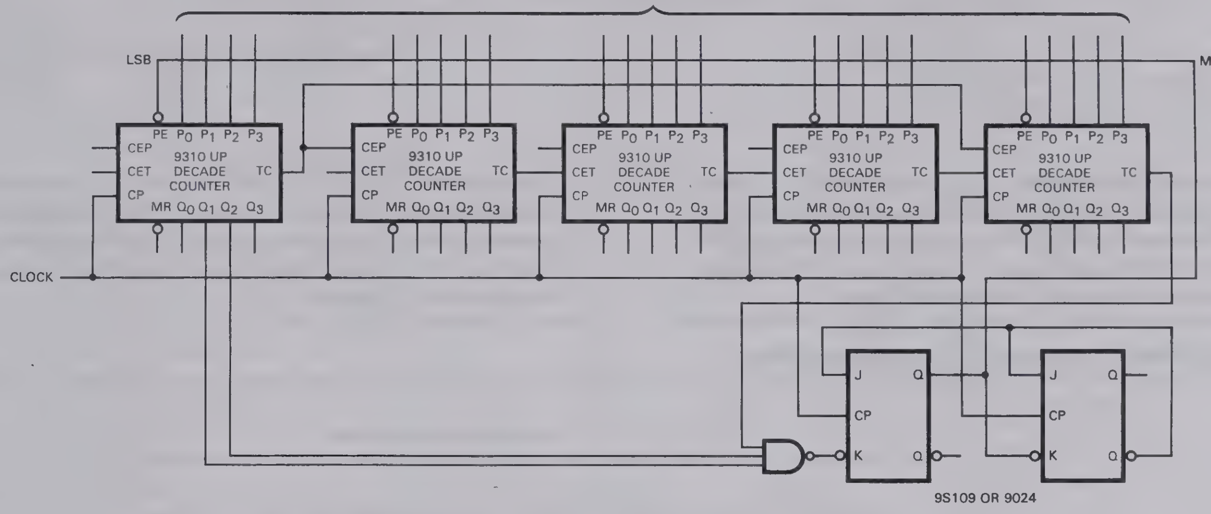
The clock pulse that increments the counter to TC-I also resets this flip-flop, thus activating the PE input. The next clock pulse loads the counter with the program value. Guaranteed count frequency can be as high as 25 MHz, limited only by the sum of the t_{pd} of the flip-flop plus the setup time of the $\overline{PE}$ inputs.

9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

FAST MULTISTAGE PROGRAMMABLE COUNTERS WITHOUT PROGRAM RESTRICTIONS

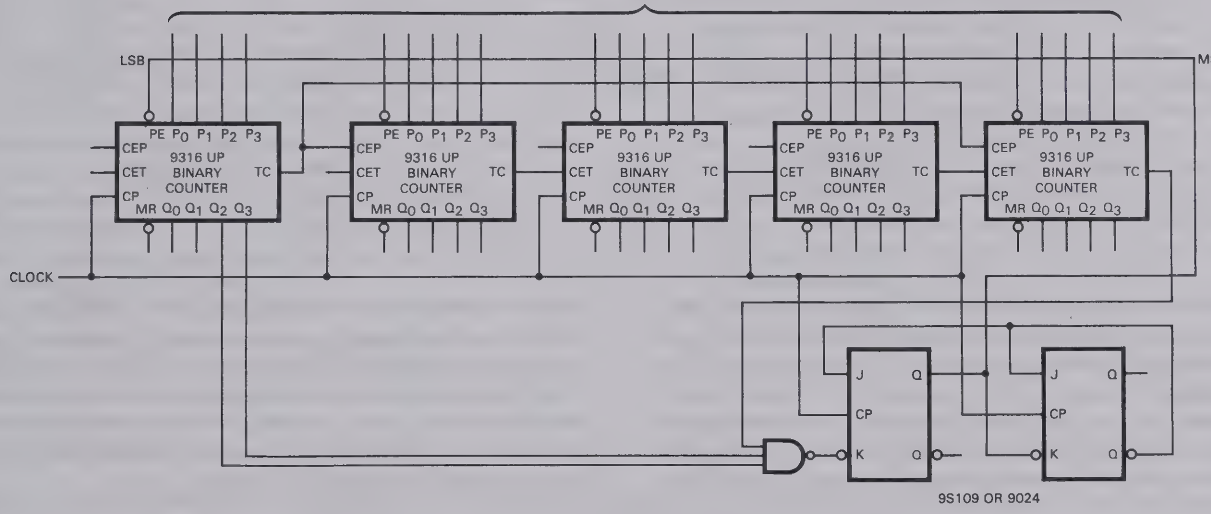
DECIMAL

PROGRAM WITH NINES COMPLEMENT


LSB
MSB
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
J
Q
CP
K
Q
9S109 OR 9024

BINARY

PROGRAM WITH ONES COMPLEMENT


LSB
MSB
PE P0 P1 P2 P3
CEP
CET
CP
9316 UP BINARY COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
J
Q
CP
K
Q
9S109 OR 9024

The programmable counters described in the previous two figures suffer from a decrease in maximum counting speed when they are programmed with certain unfavorable numbers which do not allow enough time for the decay of the TC ripple chain.

For example, assume that a BCD counter is programmed for modulo 90. The counting sequence is:

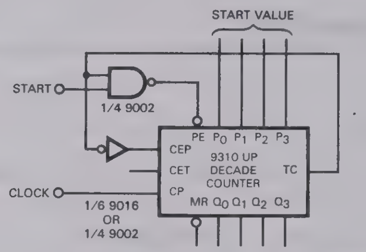
MSD	99996	LSD
	99997	
	99998	activates $\overline{PE}$
	99909	nines complement of 90 is loaded
	99910	
	etc.	

The ripple TC output from the MSD must disappear during one clock period (when 99909 has been loaded). If the clock period is shorter than this ripple delay, the next clock pulse reloads and the counter divides by the wrong number. This problem is overcome by a second flip-flop, as shown above.

The dual flip-flop provides additional time for the TC outputs to ripple Low, since it activates the $\overline{PE}$ signal for two clock pulses instead of one. The two flip-flops form a modulo 3 counter and are normally set. TC-3 is decoded and activates the reset ($\overline{K}$) input of the first flip-flop. The next clock pulse increments the counter to TC-2 and resets the first flip-flop. This activates the $\overline{PE}$ inputs and the reset ($\overline{K}$) input of the second flip-flop. The next clock pulse loads the program value into the counter and resets the second flip-flop. Since the first flip-flop remains reset, the following clock pulse loads the counter again and it sets both flip-flops. The next clock pulse increments the counter.

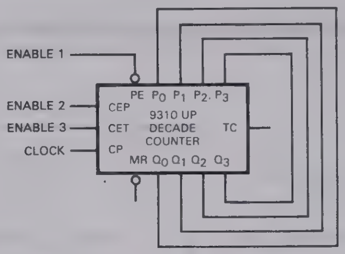
9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

SELF-STOPPING COUNTER


START VALUE
START
1/4 9002
CLOCK
1/6 9016
OR
1/4 9002
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP
DECADE
COUNTER
TC
MR Q0 Q1 Q2 Q3

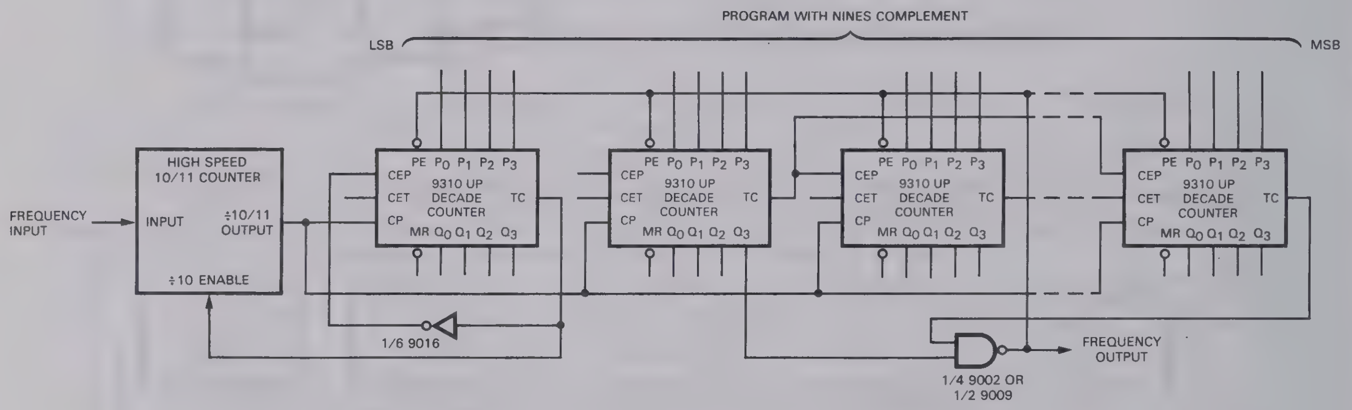
An inverter between the TC output and the CEP Enable input is all that is needed to stop a counter at its terminal state. The inverter, however, must be connected to CEP, not to the CET Enable input. Since TC is a function of CET, such a connection causes oscillation.

COUNTER WITH THREE ENABLES


ENABLE 1
ENABLE 2
ENABLE 3
CLOCK
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP
DECADE
COUNTER
TC
MR Q0 Q1 Q2 Q3

If a particular application does not require the programmable feature but does require three Enables on a counter, each Q output should be tied to the corresponding P input as shown, permitting the $\overline{PE}$ input to function as an enable.

HIGH SPEED PROGRAMMABLE DIVIDER STAGE


PROGRAM WITH NINES COMPLEMENT
LSB
MSB
FREQUENCY
INPUT
HIGH SPEED
10/11 COUNTER
INPUT
÷10/11
OUTPUT
÷10 ENABLE
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP
DECADE
COUNTER
TC
MR Q0 Q1 Q2 Q3
1/6 9016
1/4 9002 OR
1/2 9009
FREQUENCY
OUTPUT

The 9310 and 9316 can control a higher speed counter that is one of the stages of a programmable divider, extending the operating frequency by a factor of 10. A selectable divide-by-10 or divide-by-11 high speed counter acts as the clock source for the subsequent counter stages and for a control counter. The control counter and 10/11 counter form the units counter of the programmable divider. After loading the four 9310s, the control counter enables the divide-by-11 mode of the high speed counter for the duration of the units cycle. When the units cycle is over, the control counter enables the divide-by-10 mode, causing the high speed counter to become a divide-by-10 counter functioning as a normal decade units counter stage.

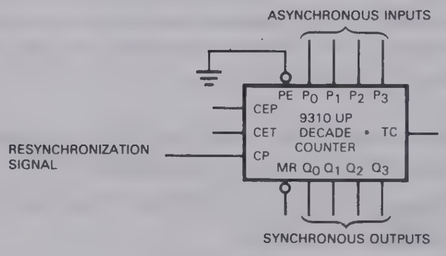
UP/DOWN COUNTERS

The 9310 and 9316 combined with gating circuitry can be used as up/down counters, but the 9360/74192 and 9366/74193 up/down counters are generally move applicable for this function.

SPECIAL APPLICATIONS

The synchronous parallel load capability of the 9310 and 9316 enables these counters to be used in applications not customarily associated with counters. For example, if the $\overline{PE}$ input of the 9310 (or 9316) is permanently grounded, the device behaves like four dual-rank D-input flip-flops, decreasing the number of different parts needed in certain systems.

RESYNCHRONIZER

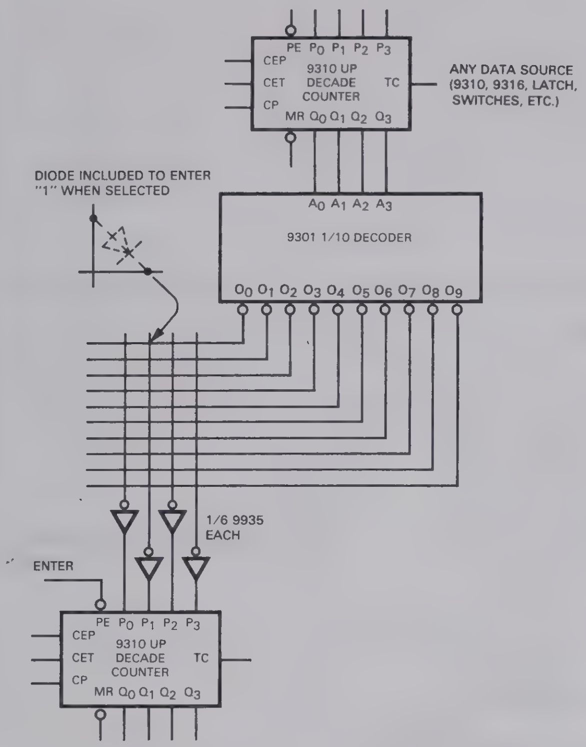

ASYNCHRONOUS INPUTS
RESYNCHRONIZATION
SIGNAL
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP
DECADE
COUNTER
*
TC
MR Q0 Q1 Q2 Q3
SYNCHRONOUS OUTPUTS

*OR 9316

A resynchronizer using a 9310 (or 9316) as four D-input flip-flops is shown. In this circuit the $\overline{PE}$ input is grounded, and the resynchronizing input is applied to the CP input. In most cases, the 9300 universal shift register is preferable for this function.

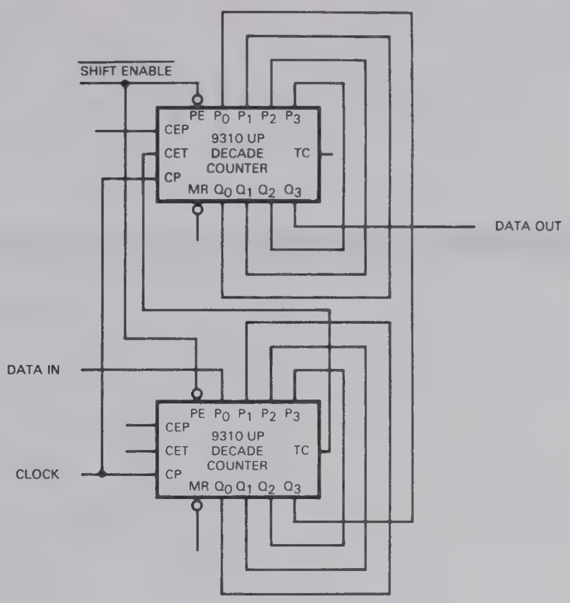
9310/93L10 AND 9316/93L16 COUNTER APPLICATIONS

PROGRAMMING COUNTERS


PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
ANY DATA SOURCE (9310, 9316, LATCH, SWITCHES, ETC.)
DIODE INCLUDED TO ENTER "1" WHEN SELECTED
A0 A1 A2 A3
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
1/6 9935 EACH
ENTER

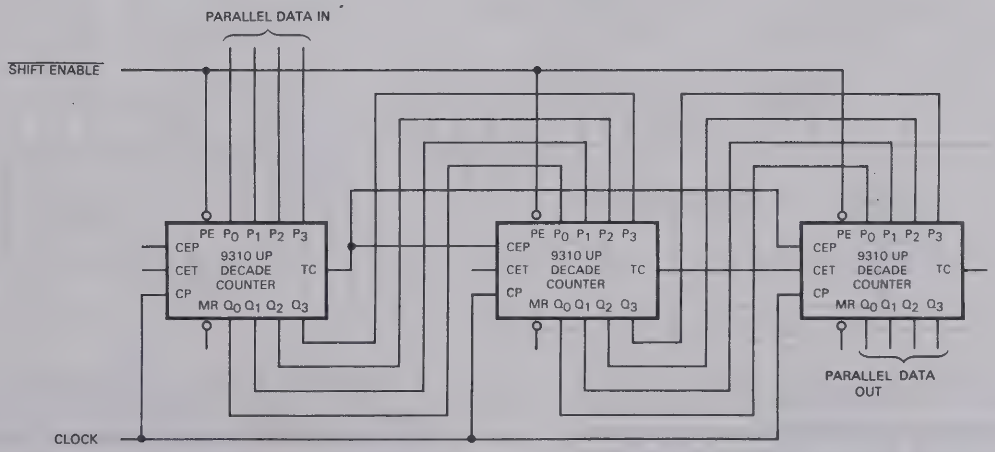
A general scheme for programming a 9310 (or 9316) counter is shown here. A diode matrix used as a read only memory or a decoding matrix (depending on how it is viewed) generates the desired code to be entered in the 9310 (or 9316). A 9301 or 9311 decoder selects the proper word, and 9935 inverters present this word to the parallel inputs. These inverters (standard DTL inverters without input diodes) are needed to maintain noise immunity on the inputs.

COMBINATION COUNTER/SERIAL REGISTERS


SHIFT ENABLE
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
DATA OUT
DATA IN
CLOCK

The contents of a 9310 or 9316 can be shifted out serially. This circuit shows each 9310 (or 9316) with Q outputs connected to the next more significant P input for shifting the most significant bit first. When the Shift Enable is High, the 9310s (or 9316s) perform the normal counting operation. When the Shift Enable is activated (Low), the contents of the two decade counters are shifted out serially at the Q_3 output of the second decade counter. If each 9310 (or 9316) output is connected to the next less significant parallel input, the least significant bit is shifted out first. Q_0 of the least significant stage provides the serial output.

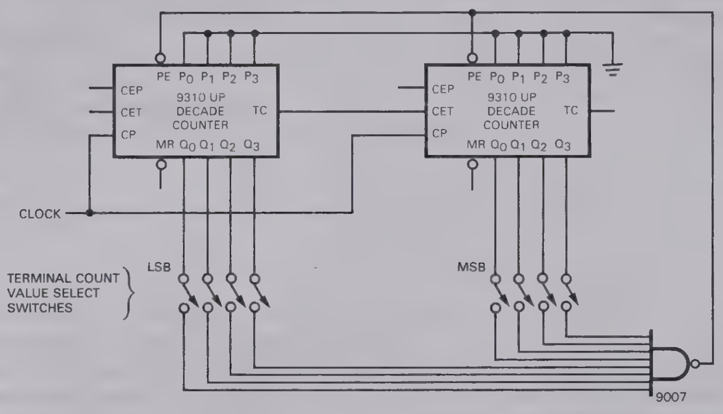
COMBINATION COUNTER/DECADE REGISTER

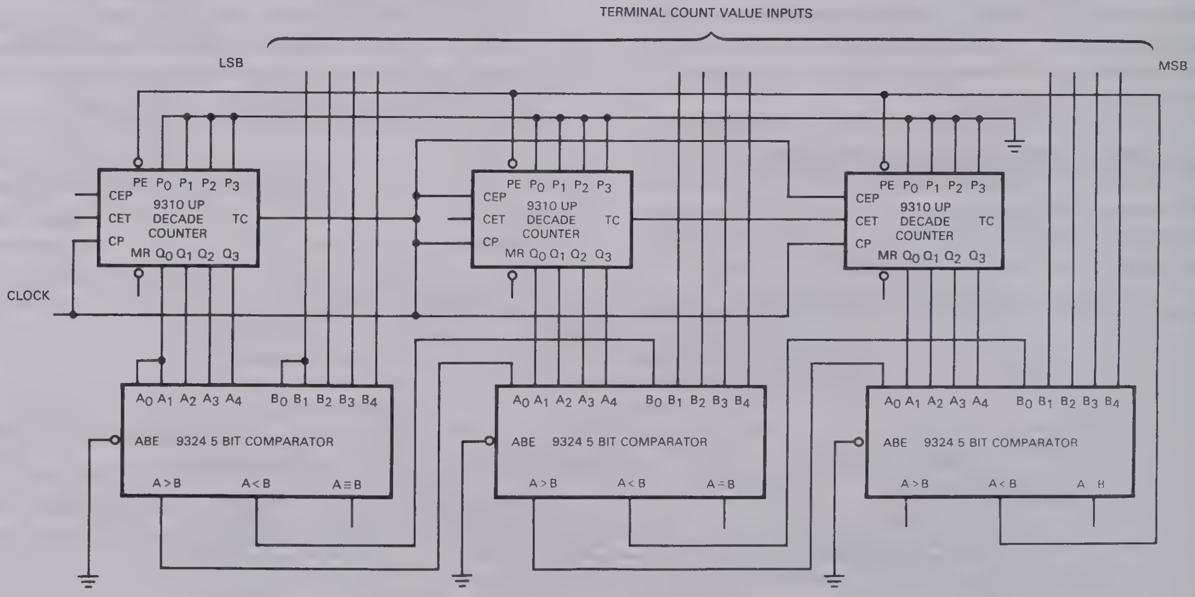

PARALLEL DATA IN
SHIFT ENABLE
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
PARALLEL DATA OUT
CLOCK

The decade counter illustrated shifts data out of each counter in parallel by decade, with the most significant decade shifted first. To shift the least significant decade first, each output is connected to the next less significant decade parallel inputs.

PROGRAMMABLE COUNTERS STARTING AT ZERO

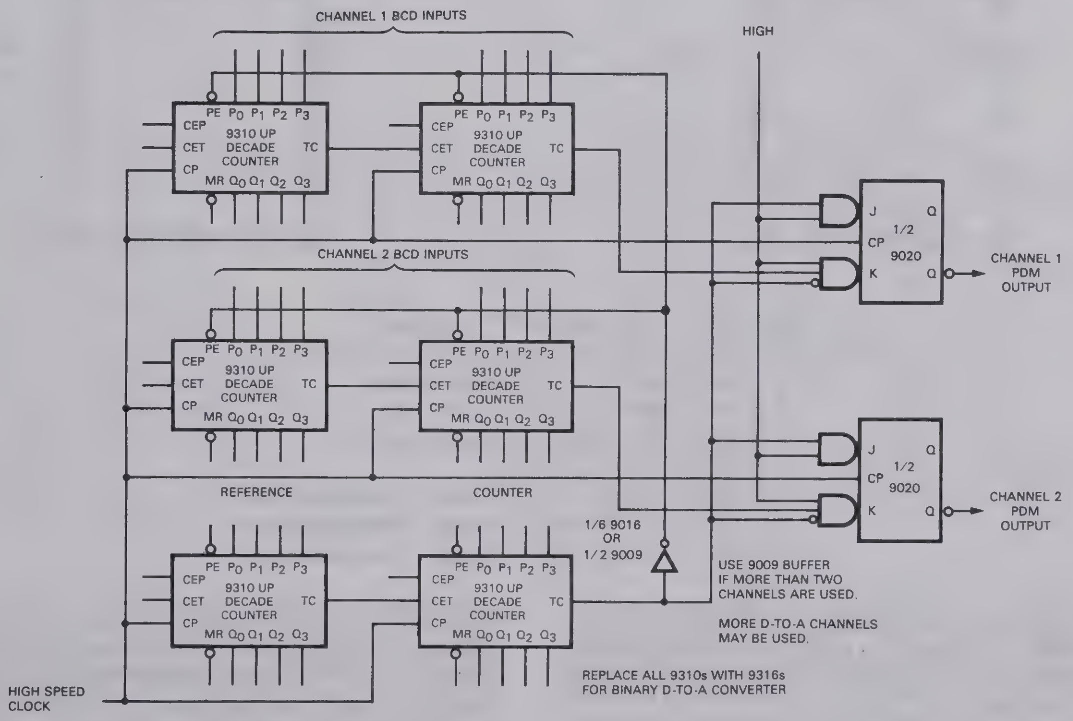
SWITCH PROGRAMMABLE


PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
LSB
MSB
TERMINAL COUNT VALUE SELECT SWITCHES
9007


TERMINAL COUNT VALUE INPUTS
LSB
MSB
PE P0 P1 P2 P3
CEP
CET
CP
9310 UP DECADE COUNTER
TC
MR Q0 Q1 Q2 Q3
CLOCK
A0 A1 A2 A3 A4
B0 B1 B2 B3 B4
ABE
9324 5 BIT COMPARATOR
A>B
A<B
A=B

Some applications call for counters that start at zero and reset at a predetermined number. This number may be derived from a switch position, from internal logic signals, or it may be permanently fixed. Two such counters are shown. In the circuit on top, a NAND gate decodes the terminal count, and on the next clock pulse the decoded output resets the counter through the $\overline{PE}$ inputs. The $\overline{PE}$ inputs make it possible to reset the counters synchronously by grounding all the P inputs. If the asynchronous $\overline{MR}$ input were used to reset the counter, a race condition would result. In the second circuit 9324 comparators compare the count value against a value generated by logic circuits. The A<B output of the comparator serves as a preset signal to the counters.

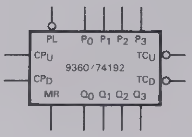
CYCLIC D/A CONVERSION

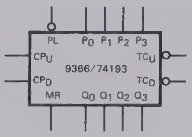

CHANNEL 1 BCD INPUTS
HIGH
CHANNEL 2 BCD INPUTS
PE P0 P1 P2 P3
CEP
CET
CP
TC
MR Q0 Q1 Q2 Q3
9310 UP DECADE COUNTER
J
K
CP
Q
1/2 9020
CHANNEL 1 PDM OUTPUT
CHANNEL 2 PDM OUTPUT
REFERENCE
COUNTER
1/6 9016 OR 1/2 9009
USE 9009 BUFFER IF MORE THAN TWO CHANNELS ARE USED.
MORE D-TO-A CHANNELS MAY BE USED.
REPLACE ALL 9310s WITH 9316s FOR BINARY D-TO-A CONVERTER
HIGH SPEED CLOCK

This is a schematic of a dual D/A PDM converter using 9310s (or 9316s). This circuit uses one programmable counter per channel plus one reference counter. The number to be converted is supplied to the parallel inputs of the programmable counter and entered when the reference counter is at Terminal Count (TC active). The programmable counter count value is greater than the count value of the reference counter by a number of counts equal to the digital input. To produce a PDM output directly proportional to the phase difference of the two counters, the Terminal Counts of the programmable and reference counters alternately set and reset the 9020 flip-flop. Additional D/A conversion channels are easily obtained by adding a programmable counter and flip-flop for each channel desired. An alternate D/A converter (more economical for multichannel conversion), using the 9318 priority encoder is included in encoders.

9360/74192, 9366/74193 UP/DOWN COUNTERS

DESCRIPTION



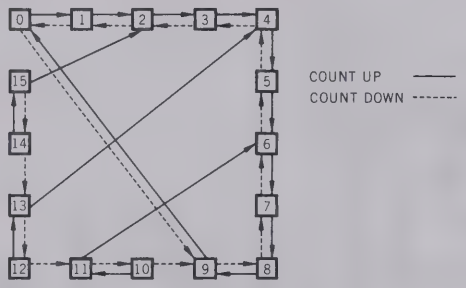


LEADS		LOADING
$\overline{PL}$	Parallel Load (Active Low) Input	1 UL
P_0, P_1, P_2, P_3	Parallel Data Inputs	1 UL
CP_U	Count Up Clock Pulse Input	1 UL
CP_D	Count Down Clock Pulse Input	1 UL
MR	Master Reset (Clear) Input (Asynchronous)	1 UL
Q_0, Q_1, Q_2, Q_3	Counter Outputs	10 UL
$\overline{TC}_U$	Terminal Count Up (Carry) Output	10 UL
$\overline{TC}_D$	Terminal Count Down (Borrow) Output	10 UL

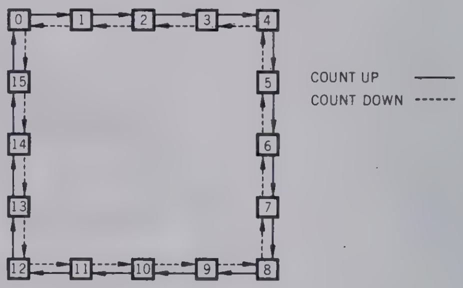
The 9360/74192 is an up/down decade (8421) counter and the 9366/74193 is an up/down 4-bit binary counter. Both devices are synchronous dual-clock up/down counters with asynchronous Parallel Load, asynchronous overriding Master Reset and internal Terminal Count logic which allows the counters to be easily cascaded without additional logic. The 9360 and 9366 can be used in many up/down counting applications, particularly when the initial count value must be loaded into the counter and multistage counting is required. The counters have active pullups for high speed, input clamp diodes to minimize the effect of line reflections, excellent noise margins and are compatible with all members of the Fairchild TTL family.

OPERATING MODES AND COUNT SEQUENCES

9360/54192, 74192 STATE DIAGRAM



9366/54193, 74193 STATE DIAGRAM



MR	$\overline{PL}$	CP_U	CP_D	MODE
L	H	H	H	No Change
L	H	CP	H	Count Up
L	H	H	CP	Count Down
L	L	X	X	Preset (Asynchronous)
H	L	X	X	Reset (Asynchronous)
H	H	X	X	Reset (Asynchronous)

The 9360 and 9366 can be Reset, Preset and can count up and down. The operating modes of the counters are listed and are identical. The only difference is in their count sequences.

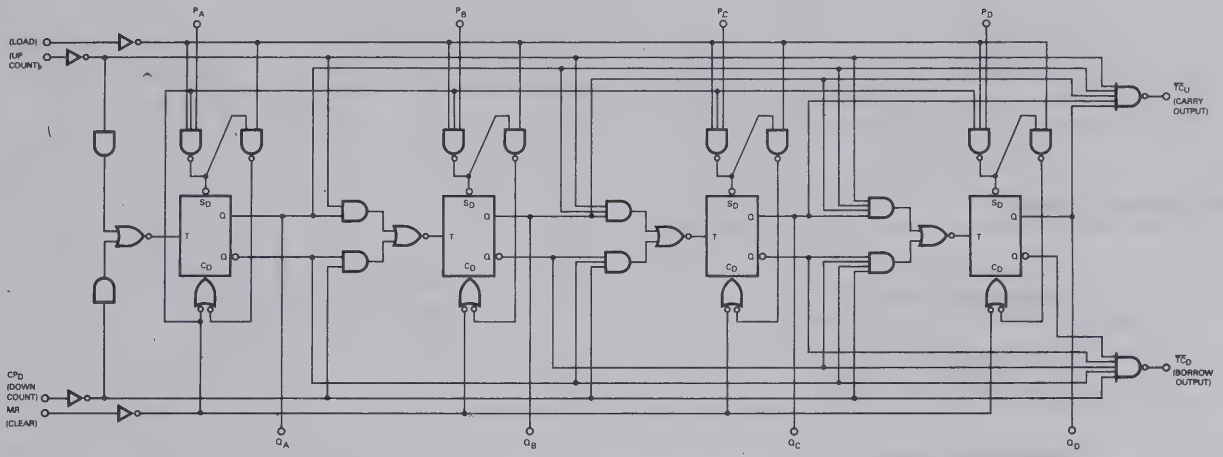
Counting is synchronous with the outputs changing state after the Low-to-High transition of either the count up clock (CP_U) or count down clock (CP_D). The direction of the count is determined by the clock input which is pulsed while the other count input is High.

Incorrect counting can occur if both the count up clock and count down clock inputs are Low simultaneously. The counters respond to a clock pulse on either input by changing to the next appropriate state of the sequences shown above. The 9360 diagram shows the regular BCD (8421) sequence as well as the sequence of states if a code greater than nine is preset into the counter.

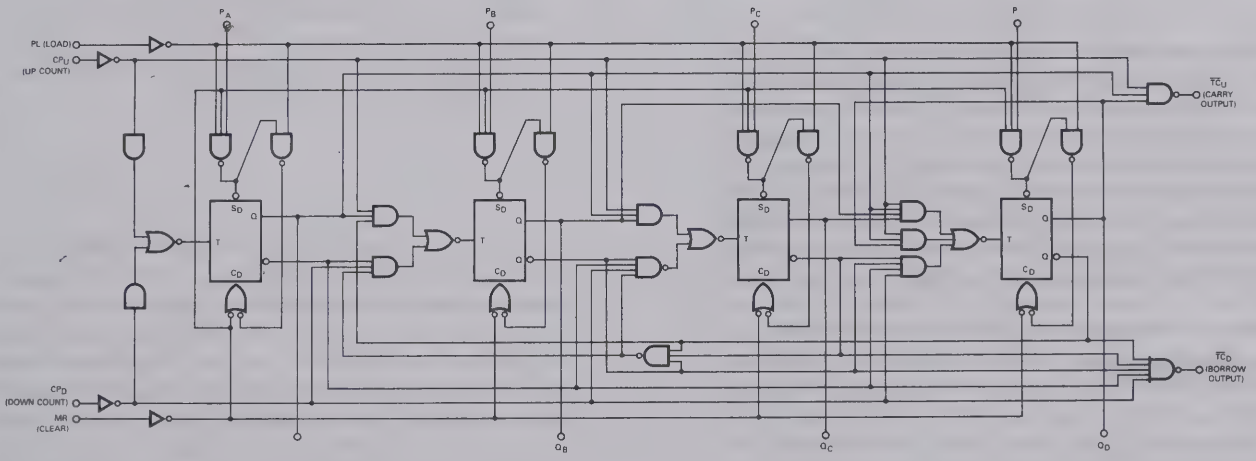
The 9360 and 9366 have an asynchronous parallel load capability permitting the counter to be preset. When the Parallel Load ($\overline{PL}$) and the Master Reset (MR) inputs are Low, information present on the parallel data inputs (P_0, P_1, P_2 and P_3) is loaded into the counter and appears on the outputs regardless of the conditions of the clock inputs. When the Parallel Load input goes High, this information is stored in the counter and when the counter is clocked it changes to the next appropriate state in the count sequence. The parallel inputs are inhibited when the Parallel Load is High and have no effect on the counter. A High on the asynchronous Master Reset (MR) input overrides both clocks and Parallel Load and clears the counter. Obviously, for predictable operation, the Parallel Load and Master Reset must not be deactivated simultaneously.

OPERATION

9360


P_A
P_B
P_C
P_D
(LOAD)
(UP COUNT)
$\overline{TC}_U$ (CARRY OUTPUT)
S_D
Q
T
C_D
CP_D (DOWN COUNT)
MR (CLEAR)
$\overline{TC}_D$ (BORROW OUTPUT)
Q_A
Q_B
Q_C
Q_D

9366


P_A
P_B
P_C
P
PL (LOAD)
CP_U (UP COUNT)
$\overline{TC}_U$ (CARRY OUTPUT)
S_D
Q
T
C_D
CP_D (DOWN COUNT)
MR (CLEAR)
$\overline{TC}_D$ (BORROW OUTPUT)
Q_B
Q_C
Q_D

Both the 9360 and 9366 have four master/slave flip-flops plus steering, Terminal Count decoding and Preset logic. Each flip-flop is designed to toggle after each clock pulse. Counting occurs by steering clock pulses from either the up or down clock input to the appropriate flip-flops and output changes are coincident, two gate delays after the rising clock edge.

The steering logic in the 9366 allows a particular flip-flop to receive an up clock pulse when all preceding stages are one and to receive a down clock pulse when all preceding stages are zero. The first flip-flop toggles if an up or down clock is received. The 9360 incorporates slightly different steering logic to allow decade counting. Each flip-flop is a master/slave toggle flip-flop operating as follows. When the toggle clock input is Low, the slave is steady but the master is set to the opposite state of the slave. During the Low-to-High clock transition, the master is disabled so a later change in the slave outputs does not affect the master. Also, the information now in the master is transferred to the slave and appears at the output. When the transfer is completed, the master and slave are steady as long as the clock input remains High. During the High-to-Low clock transition, the transfer path from master to slave is inhibited, leaving the slave steady in its present state and allowing the master to be set to the opposite state of the slave.

Asynchronous Set and Clear inputs on each flip-flop allow the respective flip-flops to be set or cleared independently of the clock inputs. All inputs are buffered and require only one unit load of drive. All outputs have drive capability of 10 unit loads. The input loading and drive capability of this device reduce the need for additional external buffers in digital systems.

TERMINAL COUNT LOGIC

9360/54192, 74192

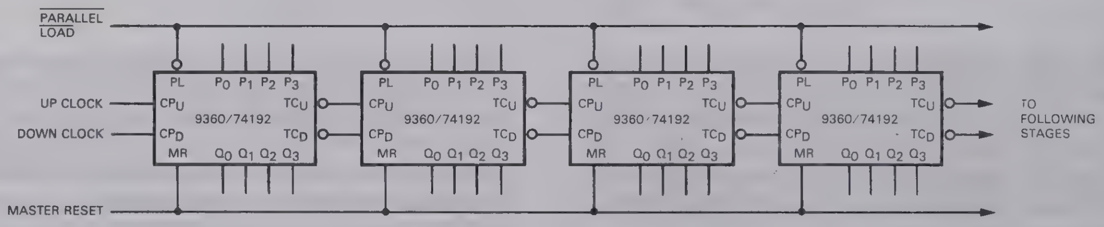
$$TC_U = Q_0 \bullet \overline{Q}_1 \bullet \overline{Q}_2 \bullet Q_3 \bullet \overline{CP}_U$$
$$TC_D = \overline{Q}_0 \bullet \overline{Q}_1 \bullet \overline{Q}_2 \bullet \overline{Q}_3 \bullet \overline{CP}_D$$

9366/54193, 74193

$$TC_U = Q_0 \bullet Q_1 \bullet Q_2 \bullet Q_3 \bullet \overline{CP}_U$$
$$TC_D = \overline{Q}_0 \bullet \overline{Q}_1 \bullet \overline{Q}_2 \bullet \overline{Q}_3 \bullet \overline{CP}_D$$

The 9360 and 9366 have Terminal Count Up ($\overline{TC}_U$) and Terminal Count Down ($\overline{TC}_D$) outputs which allow multistage ripple binary and decade counter operations without additional logic. The Terminal Count Up output is Low while the up clock input is Low and the counter is in its highest state (15 for the 9366, nine for the 9360). Similarly, the Terminal Count Down output is Low while the down clock input is Low and the counter is in state zero. The logic equations for Terminal Count are given here.

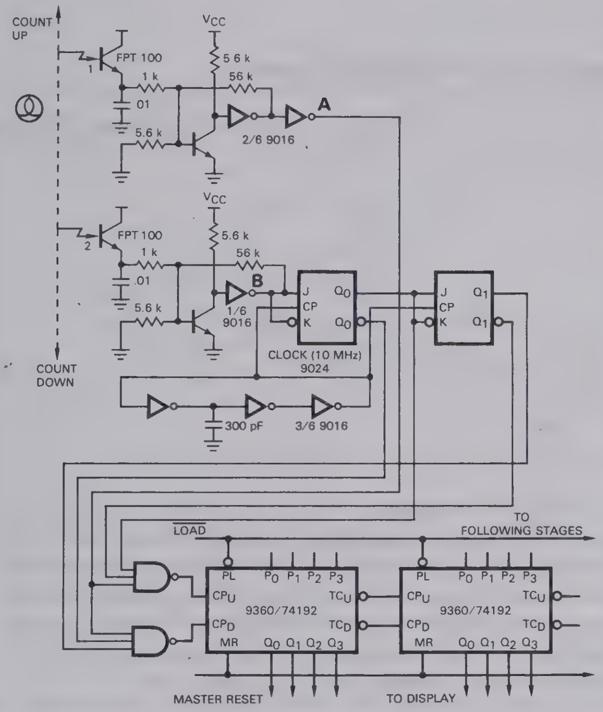
COUNTER CASCADING



The counters are cascaded by feeding the Terminal Count Up output to the up clock input and the Terminal Count Down output to the down clock input of the following (more significant) counter. Therefore, when a 9366 counter is in state 15 and counting up or in state zero and counting down, a clock pulse will change the counter's state on the rising edge and simultaneously clock the following counter through the appropriate active Low terminal output. The operation of the 9360 is the same, except when counting up, clocking occurs on state nine. The delay between the clock input and the Terminal Count output of each counter is two gate delays (typically 18 ns). Obviously these delays are cumulative when cascading counters. When a counter is reset, the Terminal Count Down output goes Low if the down clock is Low and, conversely, if a counter is preset to its terminal count value, the Terminal Count Up output goes Low while the clock is Low.

9360/74192, 9366/74193 UP/DOWN COUNTER APPLICATIONS

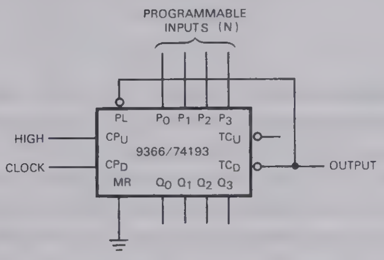
LIGHT-CONTROLLED UP/DOWN COUNTING


COUNT UP
VCC
FPT 100
1
5.6 k
1 k
56 k
.01
A
5.6 k
2/6 9016
VCC
FPT 100
2
5.6 k
1 k
56 k
.01
B
5.6 k
1/6 9016
J
CP
K
Q0
Q0
J
CP
K
Q1
Q1
CLOCK (10 MHz)
9024
COUNT DOWN
300 pF
3/6 9016
LOAD
TO FOLLOWING STAGES
PL P0 P1 P2 P3
CPU
TCU
9360/74192
CPD
TCD
MR Q0 Q1 Q2 Q3
PL P0 P1 P2 P3
CPU
TCU
9360/74192
CPD
TCD
MR Q0 Q1 Q2 Q3
MASTER RESET
TO DISPLAY

Many industrial or scientific applications require a count of objects traveling in different directions. The circuit shown counts moving objects as they move between a light source and phototransistors. This permits a count of objects passing in either direction and allows for reversals in movement or non-uniform movement. Each object passing from bottom to top increments the counter. Each object passing from top to bottom decrements the counter. Any object passing between the light source and the two phototransistors is counted as long as the object is large enough to cover both transistors simultaneously. Hex inverters serve as a clock generator and as phototransistor amplifiers. The dual flip-flop and 3-input NAND gates are used to route the phototransistors' signals to the up/down counters.

When an object moves from bottom to top it covers phototransistor 2 first, bringing line B Low. This stores a zero in the 2-bit shift register. As the object continues, phototransistor 1 is then covered and brings line A High. As the object moves even further, it uncovers phototransistor 2, bringing line B High again. The next clock pulse loads a 1 into the first bit of the shift register. This one - zero combination in the shift register and High level on line A are decoded and gated with the clock to increment the counter. For an object moving from top to bottom, the sequence is reversed and the counter decremented.

PROGRAMMABLE DIVIDER

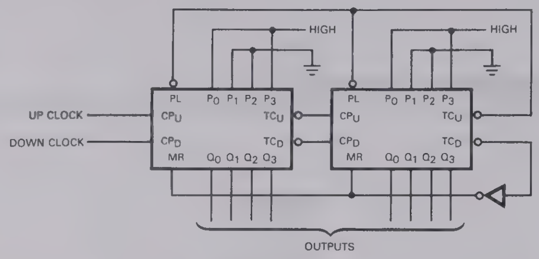

PROGRAMMABLE INPUTS (N)
HIGH
CLOCK
PL P0 P1 P2 P3
CPU
TCU
9366/74193
CPD
TCD
MR Q0 Q1 Q2 Q3
OUTPUT

DIVIDE RATIO	INPUT REQUIRED P_0	P_1	P_2	P_3	DECIMAL EQUIVALENT OF INPUT
1	H	L	L	L	1
2	L	H	L	L	2
3	H	H	L	L	3
4	L	L	H	L	4
5	H	L	H	L	5
6	L	H	H	L	6
7	H	H	H	L	7
8	L	L	L	H	8
9	H	L	L	H	9
10	L	H	L	H	10
11	H	H	L	H	11
12	L	L	H	H	12
13	H	L	H	H	13
14	L	H	H	H	14
15	H	H	H	H	15

The 9360 and 9366 can act as programmable dividers without additional logic. The divide ratio, n, is directly programmable in binary or BCD by using the count down capabilities of either counter. The divider shown operates as follows. The counter counts down until the terminal count value is reached, and when the clock goes Low again the Terminal Count output goes Low and starts to load the initial count value into the counter. When the preset number appears on the counter outputs, the Terminal Count Down output disappears and the counter decrements when the clock goes High.

The input clock width must exceed the sum of the Terminal Count delays (two gate delays per counter), the asynchronous load delays and the clock setup time. The Terminal Count output is a short pulse which should be lengthened for some applications before it is applied to other logic. It can be applied to a 9601 one-shot or to a TTL flip-flop divide-by-two stage producing a symmetrical output at half the programmed frequency. Although the Terminal Count disappears as soon as a single output changes, the internal delays are such that all flip-flops become preset before Terminal Count disappears. This is because the Preset signal must propagate through both the flip-flop and the Terminal Count gate before the preset signal is removed. Additional delay may be desired between the Terminal Count output and the Parallel Load input and is accomplished with an appropriate number of inverters.

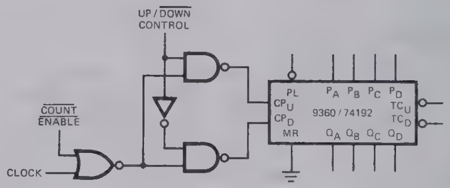
DEAD END COUNTERS



Some systems using up/down counters require that underflow or overflow be inhibited. A change from the maximum count to zero in the count up mode or from zero to the maximum count in the count down mode must be prevented. To achieve this limited range operation, the feedback connections illustrated for two 9360 decade counters are used. The same feedback can be used over any number of stages and can also be used with the 9366 binary counter. However, 15 must be loaded into the 9366 to prevent overflow.

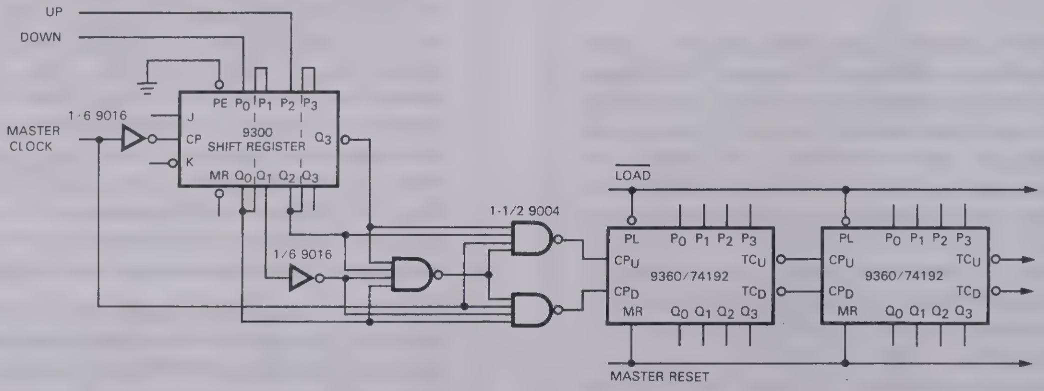
The lower limit for counting is established at zero by inverting the Terminal Count Down output and applying it to the Master Reset input. Therefore, when the counter is at state zero and a down clock is applied, Terminal Count Down activates Master Reset during the entire time the clock is Low, keeping the counter at zero and preventing the counter from decrementing. After the down clock goes High, Master Reset is removed but the counter is still in state zero. The upper limit for counting is established by connecting the Terminal Count up output to the Parallel Load input and applying the terminal count value to the preset inputs.

SINGLE LINE UP/$\overline{\text{DOWN}}$ CONTROL



This figure illustrates the addition of a single line Up/Down Control and Enable to the 9360 and 9366. All changes in the Up/Down Enable should be made while the clock is High. This restriction is not a problem since logic transitions in most of the Fairchild TTL family follow the Low-to-High transition of the clock pulse and the inputs are steady before the clock goes Low.

SYNCHRONIZATION AND COINCIDENT PULSE PREVENTION



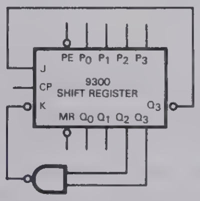
A method of synchronizing asynchronous up/down input pulses and avoiding coincident pulses to the counters is illustrated here. The counters decrement or increment when either up or down asynchronous input makes a Low-to-High transition. If both inputs make the transitions simultaneously, the counters do not decrement or increment. A master clock with a frequency of at least twice the frequency of the asynchronous inputs is needed to avoid the loss of input pulses. The 9360 counter outputs are synchronized with the master clock.

The asynchronous up/down inputs are fed to a 9300 4-bit shift register connected to form two independent 2-bit shift registers. The outputs Q_0 and Q_1 reflect the information on the asynchronous down input during two clock periods and the Q_2 and Q_3 outputs reflect information on the up input during the same clock periods. This information is decoded by the three 4-input NAND gates and gated with the clock to produce the 9360 (or 9366) up or down pulses. A Low-to-High transition on the up input results in Q_3 Low and Q_2 High. This is decoded and gated with the clock to increment the counter. A Low-to-High transition on the down input results in Q_1 Low and Q_0 High. This is decoded and gated with the clock to decrement the counter. A 4-input NAND gate disables both clocks to the counter when both transitions have occurred simultaneously.

THE 9300 SHIFT REGISTER AS A COUNTER

COUNTING WITH SHIFT REGISTERS

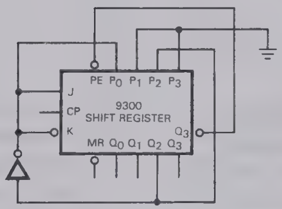
MODULO 8


PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1

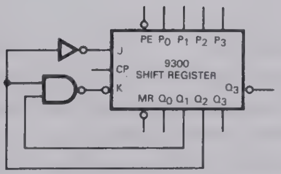
MODULO 6


PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	1
1	0	0	0
1	1	0	0
1	1	1	0
0	1	1	1
0	0	1	0

MODULO 6


PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3

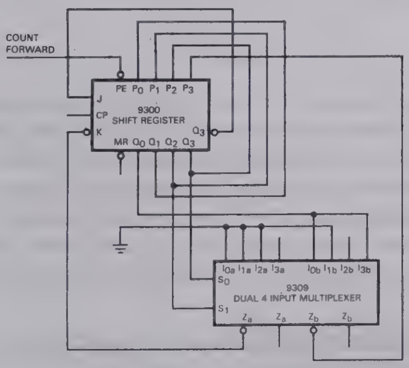
Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	1
1	0	0	0
1	1	0	0
1	1	1	0
0	1	1	1
0	0	1	1

The 9300 4-bit universal shift register can be used for a wide variety of counting circuits including simple counters of different modulo, variable modulo counters and up/down counters. Twisted ring counters offer glitch-free decoding of any individual state with one inverter and one 2-input NAND gate. Decoding any group of adjacent states (2,3,4,5,6 or 7) is equally simple. The unused states of these counters are non-persistent, i.e., the counter reverts into its operating loop if accidentally set to an unused state.

TWISTED RING (JOHNSON OR MOEBIUS) REVERSIBLE COUNTERS

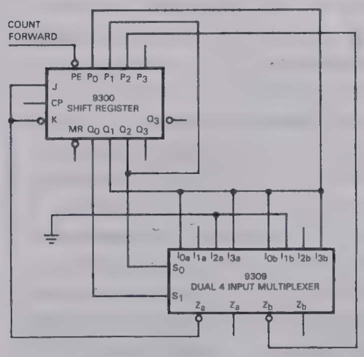
MODULO 8


COUNT
FORWARD
PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309
DUAL 4 INPUT MULTIPLEXER
Za Za Zb Zb

Count Sequence

Q_0	Q_1	Q_2	Q_3
0	0	0	0
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1

MODULO 6


COUNT
FORWARD
PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
I0a I1a I2a I3a I0b I1b I2b I3b
S0
S1
9309
DUAL 4 INPUT MULTIPLEXER
Za Za Zb Zb

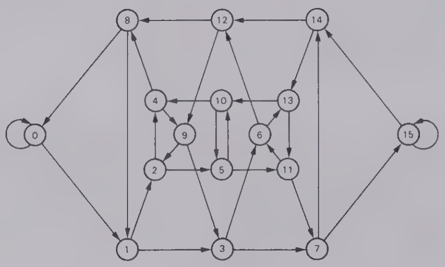
Count Sequence

Q_0	Q_1	Q_2
0	0	0
1	0	0
1	1	0
1	1	1
0	1	1
0	0	1

Twisted ring reversible counters are possible with 9300 shift registers and 9309 multiplexers. Individual or adjacent states are easily decoded without glitches with 2-input NAND gates and inverters. Again, all unused states are non-persistent.

THE 9300 SHIFT REGISTER AS A COUNTER

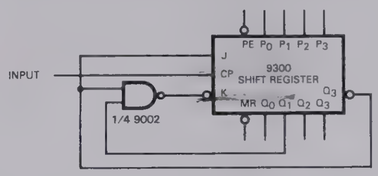
LINEAR FEEDBACK SHIFT COUNTERS

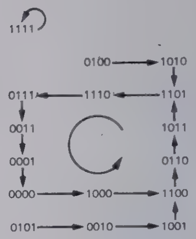

8
12
14
4
10
13
0
9
6
15
2
5
11
1
3
7

The most economical approach to the design of a counter with the 9300 is the shift counter technique. The general state diagram of a 4-bit feedback shift register is shown. Each state is identified by a decimal number which is the equivalent of the contents of the register taken as a binary number. The first bit in the register is the least significant bit. Each state has two entrances and two exits. One exit results from shifting a zero into the register and the other from shifting in a one. A wide variety of loop sequences can be chosen from the state table by assigning one and zero exits from successive states. Count sequences can be selected based on decoding requirements, simplicity of feedback logic or other system constraints.

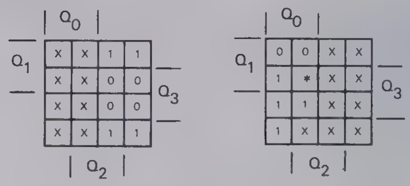
There are several feedback shift counter designs using the 9300. One approach would be to select the desired sequence of states using this general state diagram. Then, the necessary input functions would be developed to produce the selected sequence. On the first attempt to generate the input function, those states not appearing in the desired sequence (except the all zero and all one states) can be considered as "Don't Care" conditions. However, it is necessary to check the states considered as "Don't Care" conditions against the generated input function to see that they do not form any unwanted loops. The all one and all zero states are special cases since they may be persistent states (loops of one state). If an all one or an all zero state is not wanted, this requirement can be taken into consideration on the initial development of the input function by supplying a zero input for the all one state and a one for the all zero state. If the unused states form a loop, it may be necessary to modify the input function to eliminate the secondary loop. Examination of the relationship between states in the desired and undesired loops indicates the required changes in the input function for eliminating the undesired loops. Except for the 2-state loop (1010, 0101), there are several distinct loops for all loop lengths. For example, there are 16 different loop sequences of length 16. The implementation of the necessary input function is much simpler for some loops than for others.

FEEDBACK SHIFT REGISTERS


INPUT
PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
1/4 9002


1111
0100
1010
0111
1110
1101
0011
1011
0001
0110
0000
1000
1100
0101
0010
1001

J INPUT **K INPUT**

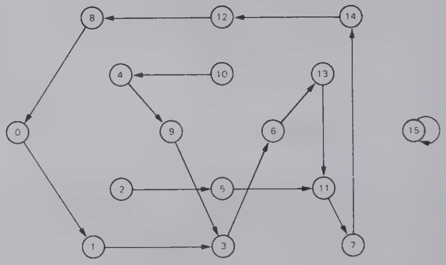

Q0
Q1
Q3
Q2
Q0
Q1
Q3
Q2

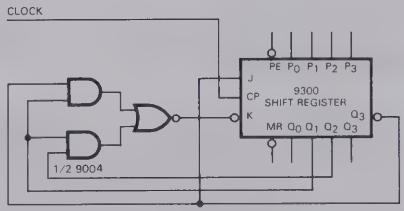
J INPUT

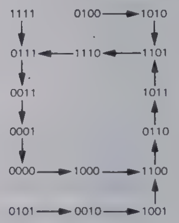
X	X	1	1
X	X	0	0
X	X	0	0
X	X	1	1

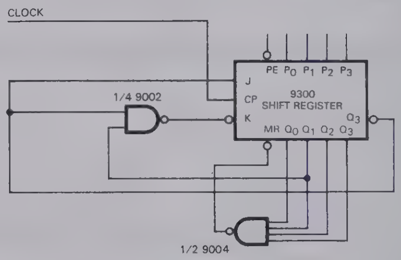
K INPUT

0	0	X	X
1	*	X	X
1	1	X	X
1	X	X	X


8
12
14
4
10
13
0
9
6
15
2
5
11
1
3
7


CLOCK
PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
1/2 9004


1111
0100
1010
0111
1110
1101
0011
1011
0001
0110
0000
1000
1100
0101
0010
1001


CLOCK
PE P0 P1 P2 P3
J
CP
K
9300
SHIFT REGISTER
Q3
MR Q0 Q1 Q2 Q3
1/4 9002
1/2 9004

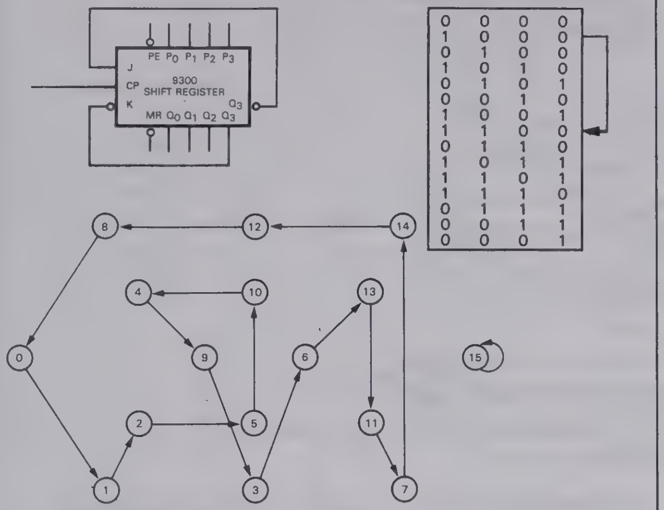
An example of this approach to the design of a feedback shift register is illustrated here. A 10-state loop is selected, and the Veitch diagrams for the J and $\overline{K}$ inputs are shown with the resulting state diagram for the simplest gate implementation. With the simplest gate implementation of the main loop, the all one state is persistent. This condition can be relieved by several approaches.

A new input function and its gate implementation results in a zero input for all one state.

Another method of eliminating the all one persistent state is to connect a 4-input gate as shown so that the all one condition produces a Master Reset input. At least one stage is then set to zero.

THE 9300 SHIFT REGISTER AS A COUNTER

MODULO 15 COUNTER

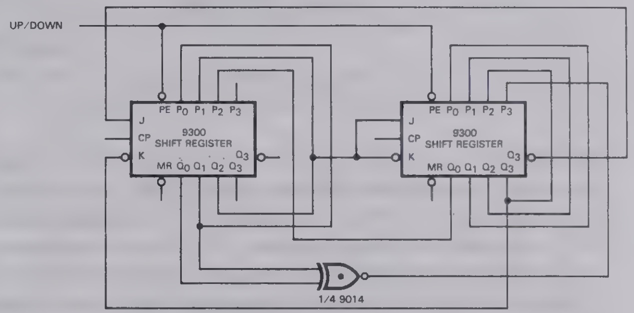

PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
8
12
14
4
10
13
0
9
6
15
2
5
11
1
3
7

JUMP WILL PRODUCE MODULO 10 COUNTER

0	0	0	0
1	0	0	0
0	1	0	0
1	0	1	0
0	1	0	1
0	0	1	0
1	0	0	1
1	1	0	0
0	1	1	0
1	0	1	1
1	1	0	1
1	1	1	0
0	1	1	1
0	0	1	1
0	0	0	1

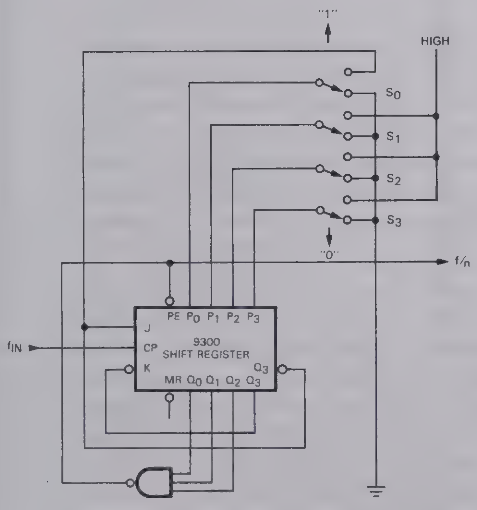
Another approach to the generation of a count sequence with a feedback shift register is to first generate a simple sequence such as one obtained by toggling the first stage of an n-bit shift register whenever the last stage is zero. This simple function produces a loop of $2^n - 1$ states and a single all one persistent state for the values of n = 2,3,4,6,7,15 and 22. Here is a 9300 connected in this manner and the resulting sequence of states. Examination of this sequence shows that a 10-state counter results if a jump is caused from the 1000 state to the 1100 state. It is only necessary to inhibit the Reset of the first state to obtain this jump. Further examination of the sequence reveals that the inhibit function is simply the Q_2 output. This approach creates exactly the same counter as obtained by the previous methods.

UP/DOWN COUNTERS

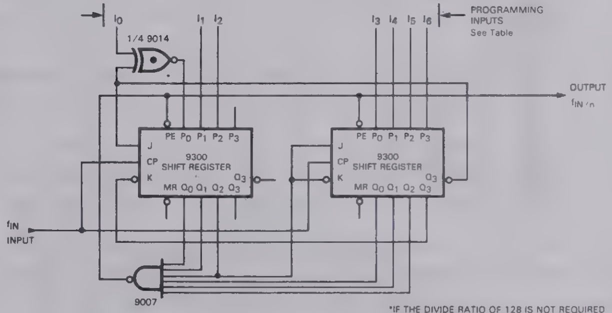

UP/DOWN
PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
1/4 9014

An up/down shift register counter is possible by connecting the 9300 as a right/left shift register (using the synchronous parallel inputs) and supplying a zero to the left shift input when the first two stages are different. This is the reverse of toggling the first stage whenever the last stage is zero. This is a modulo 127 up/down counter using this technique.

VARIABLE MODULO COUNTERS


"1"
HIGH
S0
S1
S2
S3
"0"
f/n
fIN
PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3

S_0	S_1	S_2	S_3	N
1	1	1	1	16
0	1	1	1	15
0	0	1	1	14
0	0	0	1	13
0	0	0	0	12
1	0	0	0	11
0	1	0	0	10
1	0	1	0	9
0	1	0	1	8
0	0	1	0	7
1	0	0	1	6
1	1	0	0	5
0	1	1	0	4
1	0	1	1	3
1	1	0	1	2
1	1	1	0	

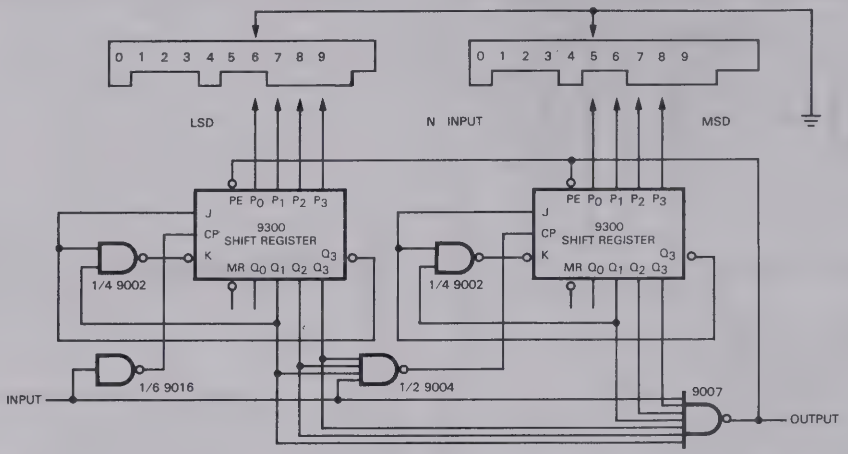

I0
I1
I2
I3
I4
I5
I6
PROGRAMMING INPUTS
See Table
1/4 9014
OUTPUT
fIN/n
PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
PE P0 P1 P2 P3
J
CP
9300
SHIFT REGISTER
K
Q3
MR Q0 Q1 Q2 Q3
fIN
INPUT
9007

*IF THE DIVIDE RATIO OF 128 IS NOT REQUIRED THIS EXCLUSIVE NOR GATE MAY BE OMITTED

Another use for the parallel input of the 9300 shift register is illustrated by the variable modulo or divide-by-n counters on the left. Again, the simple feedback (toggle the first stage when the last stage is zero) is used. This time a single gate produces the Parallel Load signal whenever the first three stages contain all ones. The parallel input combination to be loaded into the register is determined by the four switches. Note that this counter can divide by any integer up to *and including* 16. The divide-by-n counter shown on the right is simply an extension to seven bits of the 4-bit divide-by-n counter. The table opposite gives the loading values for this 7-bit counter.

THE 9300 SHIFT REGISTER AS A COUNTER

MULTISTAGE PROGRAM DIVIDER



COUNTING STATES

9	0011
8	0001
7	0000
6	1000
5	1100
4	0110
3	1011
2	1101
1	1110
0	0111

This circuit divides by any number n from 1 to 100. The selected n is one greater than is shown on the slide switches. As an example, the switches show 56, therefore the circuit will divide by 57.

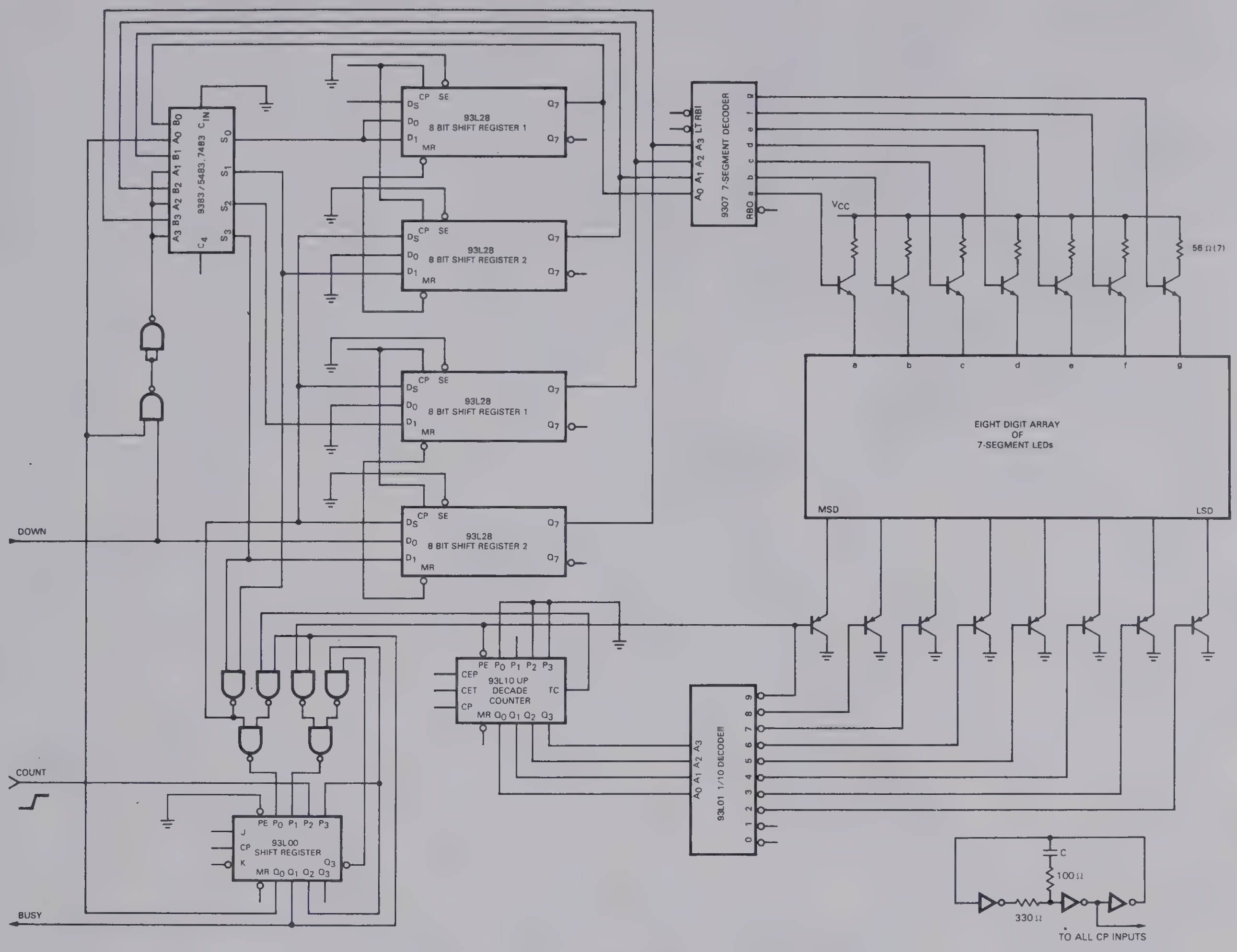
Divide-by-n counters are difficult to use in a large number of program divider applications due to the unwieldy nature of the n-input format when large values of a variable input are required. This can be overcome by building the counter in stages. This diagram is a simple, fully synchronous 2-stage decade counter. Each stage counts down modulo 10, with the first stage gating the clock to the second stage when it is in the zero condition. When both stages are in the zero condition, the Parallel Load is enabled, setting the value of n into the two decade stages. Since the decade count 00 is included in the sequence, the counter counts one more than the value of n loaded. Each additional decade stage using this approach requires either much more logic or a reduction of operating speed.

LOADING VALUES, DIVIDE-BY-n COUNTERS

n	I_0	I_1	I_2	I_3	I_4	I_5	I_6
128	1	1	1	1	1	1	1
127	0	1	1	1	1	1	1
126	0	0	1	1	1	1	1
125	0	0	0	1	1	1	1
124	0	0	0	0	1	1	1
123	0	0	0	0	0	1	1
122	0	0	0	0	0	0	1
121	0	0	0	0	0	0	0
120	1	0	0	0	0	0	0
119	0	1	0	0	0	0	0
118	1	0	1	0	0	0	0
117	0	1	0	1	0	0	0
116	1	0	1	0	1	0	0
115	0	1	0	1	0	1	0
114	1	0	1	0	1	0	1
113	1	1	0	1	0	1	0
112	0	1	1	0	1	0	1
111	0	0	1	1	0	1	0
110	1	0	0	1	1	0	1
109	1	1	0	0	1	1	0
108	0	1	1	0	0	1	1
107	0	0	1	1	0	0	1
106	0	0	0	1	1	0	0
105	1	0	0	0	1	1	0
104	0	1	0	0	0	1	1
103	0	0	1	0	0	0	1
102	0	0	0	1	0	0	0
101	1	0	0	0	1	0	0
100	0	1	0	0	0	1	0
99	1	0	1	0	0	0	1
98	1	1	0	1	0	0	0
97	0	1	1	0	1	0	0
96	1	0	1	1	0	1	0
95	0	1	0	1	1	0	1
94	0	0	1	0	1	1	0
93	1	0	0	1	0	1	1
92	1	1	0	0	1	0	1
91	1	1	1	0	0	1	0
90	0	1	1	1	0	0	1
89	0	0	1	1	1	0	0
88	1	0	0	1	1	1	0
87	0	1	0	0	1	1	1
86	0	0	1	0	0	1	1
85	0	0	0	1	0	0	1
84	0	0	0	0	1	0	0
83	1	0	0	0	0	1	0
82	0	1	0	0	0	0	1
81	0	0	1	0	0	0	0
80	1	0	0	1	0	0	0
79	0	1	0	0	1	0	0
78	1	0	1	0	0	1	0
77	0	1	0	1	0	0	1
76	0	0	1	0	1	0	0
75	1	0	0	1	0	1	0
74	0	1	0	0	1	0	1
73	0	0	1	0	0	1	0
72	1	0	0	1	0	0	1
71	1	1	0	0	1	0	0
70	0	1	1	0	0	1	0
69	1	0	1	1	0	0	1
68	1	1	0	1	1	0	0
67	0	1	1	0	1	1	0
66	1	0	1	1	0	1	1
65	1	1	0	1	1	0	1
64	1	1	1	0	1	1	0
63	0	1	1	1	0	1	1
62	0	0	1	1	1	0	1
61	0	0	0	1	1	1	0
60	1	0	0	0	1	1	1
59	1	1	0	0	0	1	1
58	1	1	1	0	0	0	1
57	1	1	1	1	0	0	0
56	0	1	1	1	1	0	0
55	1	0	1	1	1	1	0
54	0	1	0	1	1	1	1
53	0	0	1	0	1	1	1
52	0	0	0	1	0	1	1
51	0	0	0	0	1	0	1
50	0	0	0	0	0	1	0
49	1	0	0	0	0	0	1
48	1	1	0	0	0	0	0
47	0	1	1	0	0	0	0
46	1	0	1	1	0	0	0
45	0	1	0	1	1	0	0
44	1	0	1	0	1	1	0
43	0	1	0	1	0	1	1
42	0	0	1	0	1	0	1
41	0	0	0	1	0	1	0
40	1	0	0	0	1	0	1
39	1	1	0	0	0	1	0
38	0	1	1	0	0	0	1
37	0	0	1	1	0	0	0
36	1	0	0	1	1	0	0
35	0	1	0	0	1	1	0
34	1	0	1	0	0	1	1
33	1	1	0	1	0	0	1
32	1	1	1	0	1	0	0
31	0	1	1	1	0	1	0
30	1	0	1	1	1	0	1
29	1	1	0	1	1	1	0
28	0	1	1	0	1	1	1
27	0	0	1	1	0	1	1
26	0	0	0	1	1	0	1
25	0	0	0	0	1	1	0
24	1	0	0	0	0	1	1
23	1	1	0	0	0	0	1
22	1	1	1	0	0	0	0
21	0	1	1	1	0	0	0
20	1	0	1	1	1	0	0
19	0	1	0	1	1	1	0
18	1	0	1	0	1	1	1
17	1	1	0	1	0	1	1
16	1	1	1	0	1	0	1
15	1	1	1	1	0	1	0
14	0	1	1	1	1	0	1
13	0	0	1	1	1	1	0
12	1	0	0	1	1	1	1
11	1	1	0	0	1	1	1
10	1	1	1	0	0	1	1
9	1	1	1	1	0	0	1
8	1	1	1	1	1	0	0
7	0	1	1	1	1	1	0
6	1	0	1	1	1	1	1
5	1	1	0	1	1	1	1
4	1	1	1	0	1	1	1
3	1	1	1	1	0	1	1
2	1	1	1	1	1	0	1

SERIAL COUNTING WITH SHIFT REGISTERS

UP/DOWN COUNTER WITH DISPLAY MULTIPLEXER DRIVING 7-SEGMENT LED ARRAY


9383/5483, 7483
A3 B3 A2 B2 A1 B1 A0 B0
CIN
S0 S1 S2 S3
C4
93L28
8 BIT SHIFT REGISTER 1
8 BIT SHIFT REGISTER 2
DS D0 D1
CP SE
MR
Q7
9307 7-SEGMENT DECODER
A0 A1 A2 A3 LT RBI
RBO a b c d e f g
VCC
56 Ω (7)
a b c d e f g
EIGHT DIGIT ARRAY OF 7-SEGMENT LEDs
MSD
LSD
DOWN
COUNT
BUSY
93L10 UP DECADE COUNTER
PE P0 P1 P2 P3
CEP CET CP
TC
MR Q0 Q1 Q2 Q3
93L00 SHIFT REGISTER
J CP K
93L01 1/10 DECODER
0 1 2 3 4 5 6 7 8 9
C
100 Ω
330 Ω
TO ALL CP INPUTS

By separating the data storage (serial shift register) and arithmetic (adder or exclusive OR) functions, a separate class of counters develops, which is economical but slow. This approach uses four 8-bit serial shift regisrs (two 9328 dual 8-bit shift registers) to store BCD data four bits in parallel but serial by character. Incrementing or decrementing is performed by a 9383 4-bit ripple/carry adder.

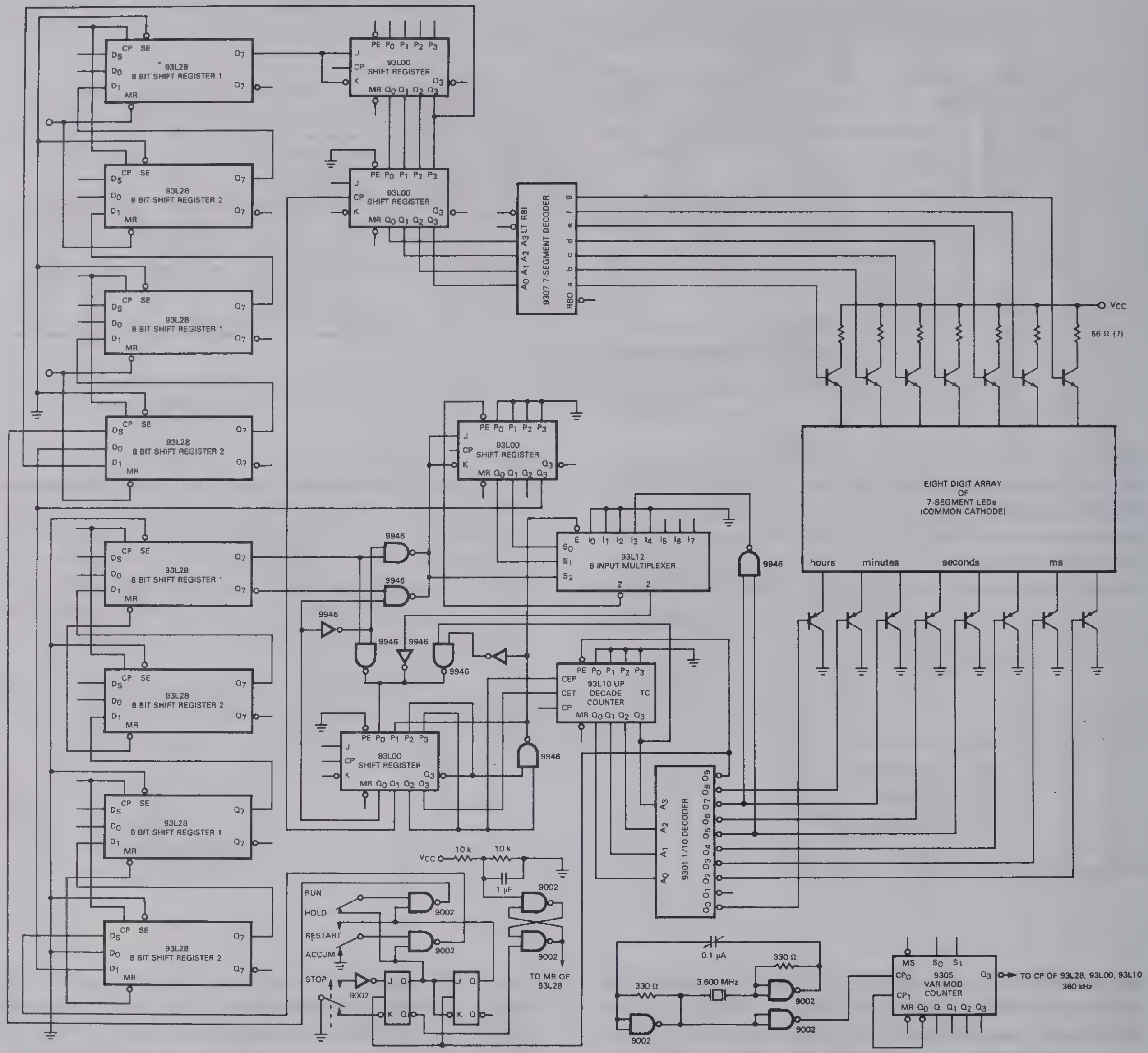
Eight BCD digits are stored in two 93L28 dual 8-bit shift registers connected as four 8-bit registers feeding through a 4-bit adder back to themselves. Counting is performed by presetting the carry flip-flop (1/4 93L00) when the least significant digit reaches the outputs of the shift registers. Whenever the carry flip-flop is set, a 1 (count up) or a 15 (count down) is added to the digit passing through the adder. BCD correction (convert 10 ⟶ 0 or 15 ⟶ 9) and generation of the carry for the next digit are performed by monitoring the adder outputs. Whenever S_1 and S_3 are High simultaneously, a carry to the next digit is generated and data is corrected by activating the other input to the shift register, where a 0 (for count up) or a 9 (for count down) is applied. A modulo 8 counter (93L10 and 93L01) keeps track of data position and controls the display position (cathodes of the LEDs). Anode information to the 7-segment LED array is generated from the outputs of the four shift registers through a 9307 7-segment decoder and seven current-limited npn drivers. The 93L00 control register performs three functions:

- Q_2 and Q_3 are connected as a 2-bit shift register that detects the rising edge of the input (count) signal (digital differentiator).
- Q_1 delays this increment or decrement instruction until it can be used to preset the carry flip-flop. (Q is brought out as an input busy signal).
- Q_0 is the carry flip-flop.

This counter counts (up or down) asynchronous input signals at a maximum rate of 1/8 the clock frequency, which in turn is defined and limited by the timing requirements of the display elements and their decoder drivers. (≈500 kHz for the LED scheme shown, only ≈10 kHz for neon discharge displays). The clock rate can be increased by using a separate scan counter and holding register and by using full power MSI devices.

SERIAL COUNTING WITH SHIFT REGISTERS

BATTERY OPERATED,CRYSTAL CONTROLLED STOPWATCH WITH MULTIPLEXED DISPLAY

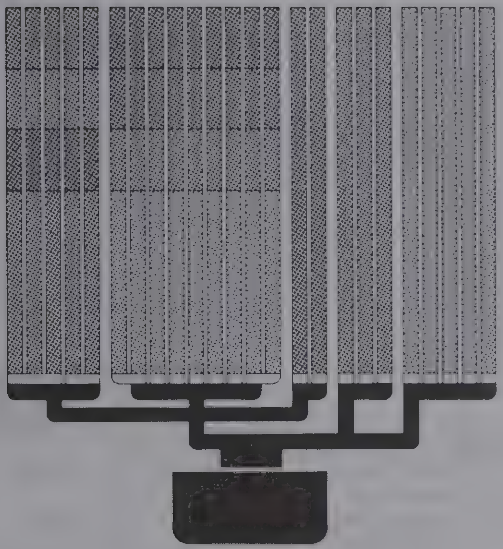


DS CP SE D0 D1 MR Q7
93L28 8 BIT SHIFT REGISTER 1
93L28 8 BIT SHIFT REGISTER 2
93L28 8 BIT SHIFT REGISTER 1
93L28 8 BIT SHIFT REGISTER 2
93L28 8 BIT SHIFT REGISTER 1
93L28 8 BIT SHIFT REGISTER 2
93L28 8 BIT SHIFT REGISTER 1
93L28 8 BIT SHIFT REGISTER 2
PE P0 P1 P2 P3 J CP K MR Q0 Q1 Q2 Q3
93L00 SHIFT REGISTER
93L00 SHIFT REGISTER
93L00 SHIFT REGISTER
93L00 SHIFT REGISTER
A0 A1 A2 A3 LT RBI RBO a b c d e f g
9307 7-SEGMENT DECODER
VCC
56 Ω (7)
EIGHT DIGIT ARRAY OF 7-SEGMENT LEDs (COMMON CATHODE)
hours
minutes
seconds
ms
9946
E I0 I1 I2 I3 I4 I5 I6 I7 S0 S1 S2 Z Z
93L12 8 INPUT MULTIPLEXER
CEP CET CP TC
93L10 UP DECADE COUNTER
9301 1/10 DECODER
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9
VCC
10 k
10 k
1 μF
9002
RUN
HOLD
RESTART
ACCUM
STOP
J Q K Q
TO MR OF 93L28
0.1 μA
330 Ω
330 Ω
3.600 MHz
MS S0 S1 CP0 CP1 Q3
9305 VAR MOD COUNTER
MR Q0 Q Q1 Q2 Q3
TO CP OF 93L28, 93L00, 93L10 360 kHz

*All 9300, 9310, 9312 and 9328 devices in this circuit are Low Power version.

This circuit stores nine digits bit-serially in a 36-bit shift register comprised of two 9328 dual 8-bit shift registers and a 9300 4-bit universal shift register incrementing or decrementing with an exclusive OR. Both this and the preceding counter offer very economical display multiplexing and are shown driving 7-segment light emitting diode displays.

The circuit above operates as a crystal-controlled stopwatch, displaying milliseconds to hours on eight 7-segment light emitting diode display devices. The time counter is a 36-bit (9-digit) bit/serial incrementer (two 93L28s, one 93L00) controlled by a 3.6 MHz crystal oscillator and time base (9305, 93L00, 93L10 and 9301) so that the 10-second and 10-minute digits are counted modulo 6. A second set of shift registers stores display data independently of the state of the counter whenever the Stop contact is activated. The contents of the storage register are strobed every four clock pulses into a 93L00 feeding a 9307 7-segment decoder. This decoder, through current-limited buffers, drives the anodes of the 8-digit LED display matrix. The cathodes are sequenced by the 93L01 and eight pnp transistors. Since this counter requires 36 clock pulses to increment the least significant digit (1/10 ms, not displayed), the shift frequency is 360 kHz, derived from a 3.6 MHz oscillator through a 9305 decade counter. In this case, the low count rate inherent to serial incrementers is advantageous, resulting in a shorter divide chain for the time base. The use of low power MSI keeps total power consumption under 2.5 W and also simplifies clock distribution.



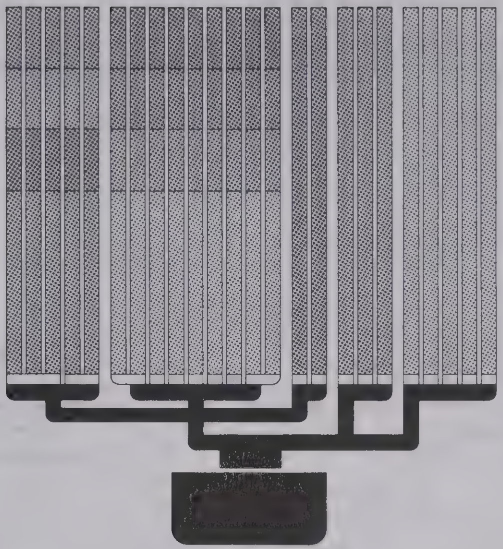
TTL Small Scale Integration

Edge Detector
Dual Edge Detector
Simple RC Clock Generator
Clock with Active Low Enable
Inexpensive Wide Range VCO
Johnson (Moebius) Counters
Switch Bounce Eliminator
Digital Differentiator
Switch Synchronizer

10
11
12
13
14
15

SSI GATE, INVERTER, BUFFER SELECTION GUIDE

	TTL 8 ns/10 mW per gate	Low Power TTL 20 ns/2 mW per gate	Standard Speed TTL 10 ns/10 mW per gate	High Speed TTL 6 ns/22 mW per gate	Super High Speed Schottky TTL 3 ns/19 mW per gate
NAND GATES					
Quad 2-Input NAND Gate	9002	9L00	9N00,54/7400	9H00,54/74H00	9S00,54/74S00
Quad 2-Input NAND Gate with Open Collector Output	9012		9N01,54/7401 9N03,54/7403	9H01,54/74H01	9S03,54/74S03
Quad 2-Input NAND Gate (15 V)			9N26,54/7426		
Triple 3-Input NAND Gate	9003		9N10,54/7410	9H10,54/74H10	9S10,54/74S10
Triple 3-Input NAND (O.C.)			9N12,54/7412		
Dual 4-Input NAND Gate	9004		9N20,54/7420	9H20,54/74H20	9S20,54/74S20
Dual 4-Input NAND Gate (O.C.)				9H22,54/74H22	9S22,54/74S22
8-Input NAND Gate	9007		9N30,54/7430	9H30,54/74H30	
13-Input NAND Gate					9S133,54/74S133
12-Input NAND Gate (3-State Output)					9S134,54/74S134
NOR GATES					
Quad 2-Input NOR Gate			9N02,54/7402		
Quad 2-2-2-4-Input NOR Gate	9015				
Triple 3-Input NOR Gate			9N27,54/7427		
Dual 4-Input NOR Gate (with strobe)			9N23-25, 54/7423-25		
AND GATES					
Quad 2-Input AND Gate			9N08,54/7408	9H08,54/74H08	
Quad 2-Input AND Gate (O.C.)			9N09,54/7409		
Quad 2-2-3-3-Input AND Gate					9S41
Triple 3-Input AND Gate			9N11,54/7411	9H11,54/74H11	9S11,54/74S11
Triple 3-Input AND Gate (O.C.)					9S15,54/74S15
Dual 4-Input AND Gate				9H21,54/74H21	
OR GATES					
Quad 2-Input OR Gate			9N32,54/7432		
EXCLUSIVE OR GATES					
Quad Exclusive OR Gate		9L86	9N86,54/7486		
Quad Exclusive OR Gate, Two Inverted Outputs	9014				
AND/OR, AND/OR/INVERT GATES AND EXPANDERS					
Dual 4-2-Input AND/OR Gate					9S42
Expandable 2-2-2-3-Input AND/OR Gate				9H52,54/74H52	
Dual 2-Wide 2-Input A/O/I Gate			9N51,54/7451	9H51,54/74H51	
Expandable Dual 2-Wide 2-Input A/O/I Gate	9005		9N50,54/7450	9H50,54/74H50	
Expandable 2-Wide, 4-Input A/O/I Gate				9H55,54/74H55	
4-Wide 2-Input A/O/I Gate		9L54	9N54,54/7454	9H54,54/74H54	
4-2-3-2-Input A/O/I Gate					9S64,54/74S64
4-2-3-2-Input A/O/I Gate (O.C.)					9S65,54/74S65
Expandable 4-Wide 2-Input A/O/I Gate			9N53,54/7453	9H53,54/74H53	
Expandable 4-Wide 2-2-2-3-Input A/O/I Gate	9008				
Triple 3-Input Expander				9H61,54/74H61	
Dual 4-Input Expander	9006		9N60,54/7460	9H60,54/74H60	
3-2-2-3-Input AND/OR Expander				9H62,54/74H62	
INVERTERS AND BUFFERS					
Hex Inverter	9016	9L04	9N04,54/7404	9H04,54/74H04	9S04-4A,54/74S04-4A
Hex Inverter with Open Collector Output	9017		9N05,54/7405	9H05,54/74H05	9S05-5A,54/74S05-5A
Hex Inverter Buffer/Driver (30 V)			9N06,54/7406		
Hex Inverter Buffer/Driver (15 V)			9N16,54/7416		
Hex Buffer/Driver (30 V)			9N07,54/7407		
Hex Buffer/Driver (15 V)			9N17,54/7417		
Quad 2-Input NAND Buffer			9N37,54/7437		
Quad 2-Input NAND Buffer (O.C.)			9N38,54/7438		
Quad 2-Input NAND Buffer (O.C., 15 V)					
Dual 4-Input NAND Buffer	9009		9N40,54/7440	9H40,54/74H40	9S40,54/74S40
Dual 4-Input NAND 50Ω Driver					9S140,54/74S140
SCHMITT TRIGGER					
Dual NAND Schmitt Trigger			9N13,54/7413		



TTL SMALL SCALE INTEGRATION

INTRODUCTION

SSI circuits (gates and flip-flops) are useful to extend logic functions available in MSI. They should be used sparingly, essentially as "glue" between MSI circuits, because they require considerably more PC board area, interconnections, power, and volume compared to equivalent functions available in MSI circuits.

GATES

Fairchild offers a wide variety of compatible gate circuits, covering different logic families (NAND, NOR, EXCLUSIVE (N)OR, AND/OR/INVERT, BUFFERS) and different semiconductor technologies (DTL, standard TTL, low power TTL, high speed TTL and very high speed TTL (Schottky). Emitter Coupled Logic (ECL) is covered separately.

The TTL technologies offer various power/speed trade-offs, while DTL and open collector TTL circuits offer the convenience of the wired-OR (collector ANDing). Typical input/output configurations, performance and power dissipation are listed in the selection guide.

The selection guide lists devices presently available. The functions are described in terms of positive logic. Obviously, a positive NAND gate can be considered a negative NOR gate and a positive NOR can be interpreted as a negative NAND.

The normal use of TTL/DTL gates is well understood and need not be covered here. The following notes describe miscellaneous uncommon circuits.

COLLECTOR-ANDed DTL GATES

When several DTL gates are collector-ANDed, (wired-OR) the original fan out for 6 kΩ DTL gates (8.5 TTL unit loads) is reduced by approximately 2/3 unit load for each additional 6 kΩ output connected to the node. The original fan out for 2 kΩ DTL gates (7.5 TTL unit loads) is reduced by approximately 2 unit loads for each additional 2 kΩ output connected to the node. Obviously, High drive capability is improved.

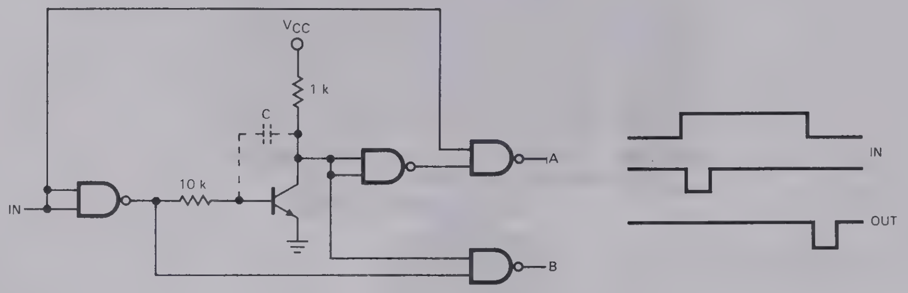
INCREASING THE VOLTAGE SWING OF TTL OUTPUTS

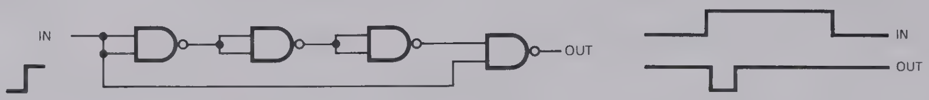
When interfacing with MOS devices or discrete transistors it may be desirable to increase the logic swing out of a TTL output, which (worst-case) is only 2.0 V. A 1 kΩ resistor connected from any TTL output to V_{CC} will pull the output close to V_{CC}, increasing the output swing to more than 4 V, while decreasing the available fan out by three unit loads.

DRIVING TRANSISTORS FROM DTL GATE OUTPUTS

Any 2 kΩ DTL gate output that does not drive any other logic can be tied directly to the base of a grounded emitter transistor to provide a worst case base drive of 1.5 mA. An additional pull up resistor of 470Ω to V_{CC} increases the base drive to 10 mA.

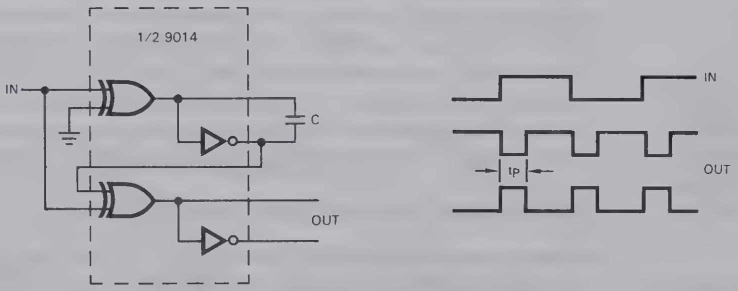
EDGE DETECTOR





This edge detector circuit generates a negative-going pulse on output A for each Low-to-High transition of the input, and generates a negative-going pulse on output B for each High-to-Low transition of the input. The pulse width is adjustable by varying the Miller capacitance. A non-adjustable short pulse (≈20ns) on the Low-to-High transition of the input can be generated by replacing the transistor inverter stage with the unused fourth NAND gate.

DUAL EDGE DETECTOR

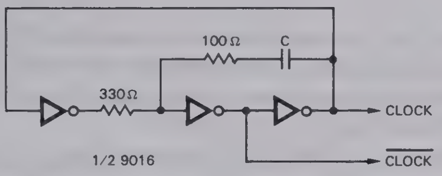


C	t_P
0	10 ns
200 pF	30 ns
1000 pF	70 ns

Half of a 9014 quad Exclusive-OR gate with one capacitor provides a circuit generating an output pulse for both a Low-to-High and a High-to-Low transition of the input signal. This function is useful for regenerating the clock in a self-clocking PDM transmission system. When fed with a square wave input, this circuit acts as a frequency doubler.

TTL SMALL SCALE INTEGRATION APPLICATIONS

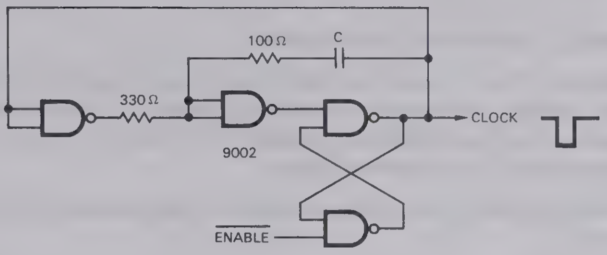
SIMPLE RC CLOCK GENERATOR



C	f
200 pF	5 MHz
1600 pF	1 MHz
0.018 μF	100 KHz
0.18 μF	10 KHz

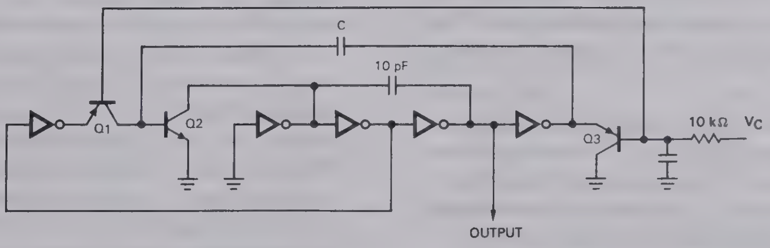
The simple TTL clock generator circuit shown provides a clock satisfactory for most simple TTL systems and it always starts oscillating without coaxing. This circuit requires only 1/2 of a hex inverter package and three passive components-two resistors and a capacitor.

ACTIVE LOW ENABLE



An active Low Enable input can be provided by replacing the 9016 with a 9002 quad 2-input NAND gate as shown above. The cross-coupled gates guarantee that clock pulses are not cut short if the Enable input is removed at the wrong time. Once the clock goes Low it stays Low for its full normal width even if the Enable signal is returned High. The clock pulse starts two gate delays (15 ns) after the High-to-Low transition of the $\overline{\text{Enable}}$ input.

INEXPENSIVE WIDE RANGE VOLTAGE CONTROLLED OSCILLATOR



The voltage controlled oscillator shown uses 5/6 of a DTL hex inverter package (9936 or 9937), three transistors and two capacitors. It can be tuned over a wide range (>30:1) because the control voltage (V_C) affects not only the voltage swing on the timing capacitor (by using Q3 as a clamp), but it also controls the amount of charging current by using Q1 and the pull up resistor of the DTL output as a controlled current source).

A low input voltage (≈0.5 V) generates a high frequency; a high input voltage (≈4 V) generates a low frequency. The 10 pF speed-up capacitor provides glitch free transitions. This circuit is limited to frequencies under 1 MHz.

SINGLE AND DUAL FLIP-FLOPS

Fairchild offers a wide variety of single and dual flip-flops with various differences in logic configuration, speed and power. Flip-flops presently available are listed in the selection guide on the following page.

There are three independent aspects to the selection of flip-flops:

- Logic configuration
- Speed and power
- Clocking scheme

LOGIC CONFIGURATION

All flip-flops (except for the 7474/74H74/74S74) are J K flip-flops; therefore, their response to any digital input condition is logically defined according to the truth table below.

J	K	Q_{n+1}
not active	not active	Q_n (i.e. remain unchanged)
active	not active	High (= SET)
not active	active	Low (= RESET)
active	active	$\overline{Q}_n$ (= TOGGLE)

(Essentially J is a Set input and K is a Reset input, but applying both J and K simultaneously toggles the flip-flop.) The flip-flops differ in the logic controlling the J and K condition. Some have no extra logic, others have a J • K Enable, some have a multi-input AND gate controlling J and K, and some have an AND/OR structure, accepting data from different sources. Note that some of the inputs are active Low. The $\overline{K}$ input is particularly useful since tying J and $\overline{K}$ together results in a D input.

All flip-flops have an asynchronous, active Low Preset or Clear input, and some of them have both. On all flip-flops except the 7470, these inputs are truly asynchronous, affecting both master and slave directly. On the 7470 the Preset and Clear inputs are active only when the clock is Low.

SPEED AND POWER

As a rough guideline, typical values for maximum clock frequency and for the delay between clock edge and output change are given in the selection guide. For more details, consult the individual product technical data sheets. Power consumption varies between 50 mW for the 9L24 (the only low power dual flip-flop) and 200 mW for the 9022.

CLOCKING SCHEME

Perhaps the most important and certainly the most confusing and misunderstood aspect of flip-flops is their clocking scheme. This defines when the flip-flop accepts data and when the outputs change with regard to the clock.

While most MSI circuits follow consistent and simple rules (data is accepted a set-up time before the rising edge of the clock, and the outputs change after the rising edge of the clock), flip-flops are not that standard.

There is a functional difference between edge-triggered and true master/slave flip-flops. In addition, some flip-flops use inverting clock buffers and others do not. As a result, there are four different clocking schemes, which are indicated in the selection guide by their clock wave shapes.

Rising-Edge Triggered

The 7470 single flip-flop and the 9024,9L24, and 7474 dual flip-flops accept data a set-up time before the rising edge of the clock and change their output after this rising edge of the clock. Clock pulse width is not critical as long as it exceeds worst case set-up time. This is consistent with the behavior of most MSI devices.

Falling-Edge Triggered

The 74H101/102 single flip-flops and the 74H103/106/108 and the 74S112/113/114 dual flip-flops accept data a set-up time before the falling edge of the clock and change their outputs after this falling edge of the clock. Their clock input is therefore shown with an inverting symbol. Interfacing with most MSI devices requires a clock inverter unless this offset triggering is desired.

Negative Clock Pulse Master/Slave

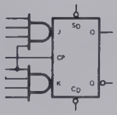
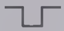
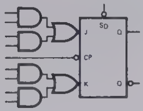
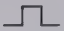
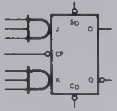
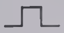
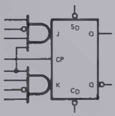
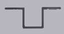
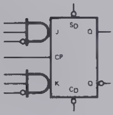
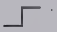
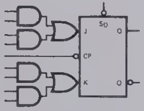
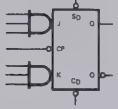
The 9000/74104 and 9001/74105 single flip-flops and the 9020/9022 dual flip-flops accept input data while the clock is Low and change their outputs after the rising edge of the clock pulse. (Of course, the clock pulse is not a negative voltage but rather a pulse going High to Low to High). The Set and Reset inputs to the master are gated with the outputs of the slave in order to achieve the JK configuration. Therefore, if an input condition has put the master into a state opposite that of the slave's during a clock pulse, nothing can change the master until the beginning of the next clock pulse. This is commonly called "ones-catching" (really "opposite state catching"), and in a conventional system it means that the clock pulse should be made as short as possible and the time between clock pulses as long as possible, for two reasons: to accommodate all ripple delays between clock pulses, and secondly, to increase ac noise immunity of the system. There are special cases where ones-catching can be used as an advantage, but in general it must be regarded as an inherent drawback of all true master/slave JK flip-flops.

Positive Clock Pulse Master/Slave

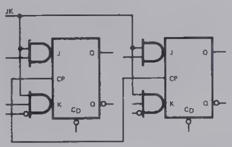
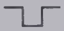
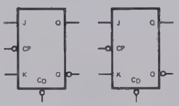
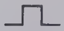
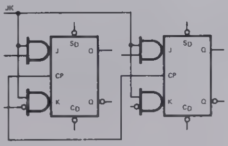
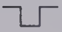
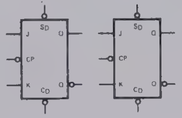
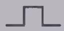
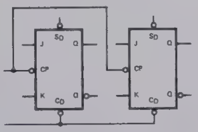
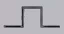
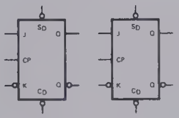
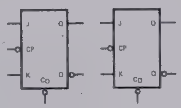
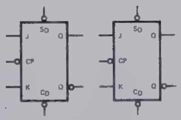
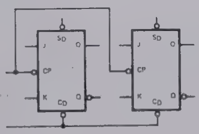
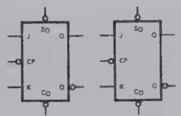
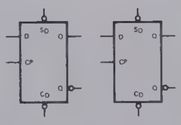
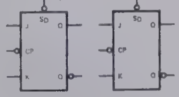
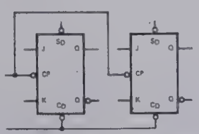
The 7472/74H22 and the 74H71 single flip-flops and the 7473/74H73, 7476/74H76, and 74H78 dual flip-flops accept input data while the clock is High and change their outputs after the falling edge of the clock pulse. As explained in the preceding paragraph, these master/slaves are ones-catching. Since their clock polarity differs from most MSI circuits, they usually require an external clock inverter.

SSI SINGLE AND DUAL FLIP-FLOP SELECTION GUIDE

SINGLE

	OUTPUT CHANGES ON POSITIVE GOING EDGE	OUTPUT CHANGES ON NEGATIVE GOING EDGE	
MASTER/SLAVE	 **9000,74104** 20 MHz/16 ns 	 **9H71,74H71** 30 MHz/22 ns 	 **9N72,7472** 20 MHz/25 ns **9H72,74H72** 30 MHz/22 ns 
	 **9001,74105** 50 MHz/16 ns 		
EDGE TRIGGERED	 **9N70,7470** 35 MHz/27 ns 	 **9H101,74H101** 50 MHz/16 ns 	 **9H102,74H102** 50 MHz/16 ns

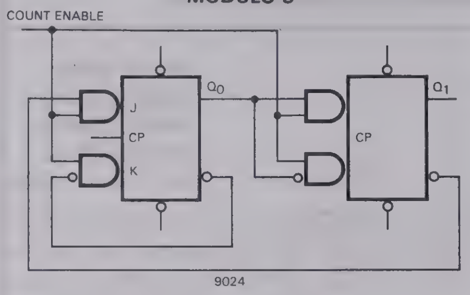
DUAL

MASTER/SLAVE	 **9020** 50 MHz/16 ns 	  **9N73,7473 • 9N107,74107** 20 MHz/25 ns **9H73,74H73** 30 MHz/22 ns	
	 **9022** 50 MHz/16 ns 	 **9N76,7476** 20 MHz/25 ns **9H76,74H76** 30 MHz/22 ns 	 **9H78,74H78** 30 MHz/22 ns 
EDGE TRIGGERED	 **9024,74109** 25 MHz/22 ns **9L24** 8 MHz/66 ns **9S109** 100 MHz/7 ns 	 **9H103,74H103** 50 MHz/16 ns 	 **9H106,74H106** 50 MHz/16 ns
		 **9H108,74H108** 50MHz/16 ns 	 **9S112,74S112** 125 MHz/5 ns
	 **9N74,7474** 25 MHz/20 ns **9H74,74H74** 43 MHz/13 ns **9S74,74S74** 100 MHz/7 ns 	 **9S113,74S113** 125 MHz/5 ns 	 **9S114,74S114** 125 MHz/5 ns

MASTER/SLAVE (or "ONES CATCHING" MASTER/SLAVE)—The master is sensitive to input conditions during the active portion of the clock pulse.
EDGE TRIGGERED — Only sensitive to input conditions immediately prior to active clock edge.

JOHNSON (MOEBIUS) COUNTERS

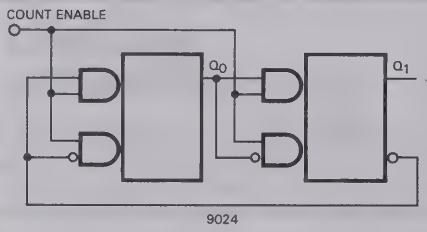
MODULO 3



COUNT SEQUENCE	
Q_0	Q_1
1	1
0	1
0	0
1	0

With this modulo 3 Johnson counter, each state can be easily decoded with a 2-input NAND gate. As shown, the unused state is non-persistent.

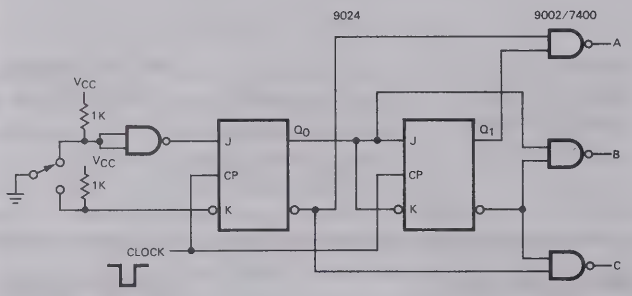
MODULO 4

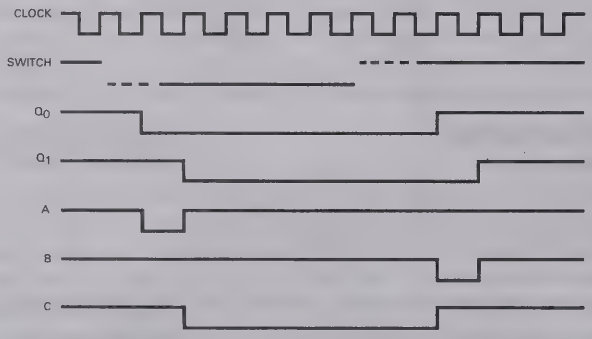


COUNT SEQUENCE	
Q_0	Q_1
0	0
1	0
1	1
0	1

The modulo 4 counter also provides for glitch free, easy decoding of each state with a 2-input NAND gate.

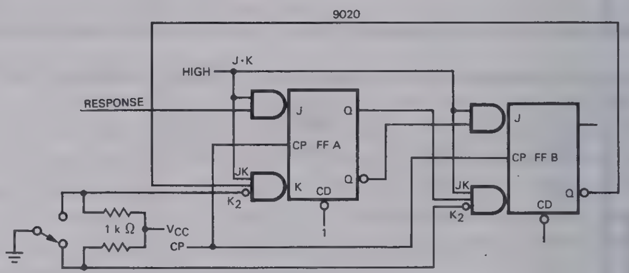
SWITCH BOUNCE ELIMINATOR AND DIGITAL DIFFERENTIATOR





This circuit eliminates switch bounce and generates active Low output pulses; one on output A after switch depression and one on output B after switch release, and a bounce-free level output C between them.

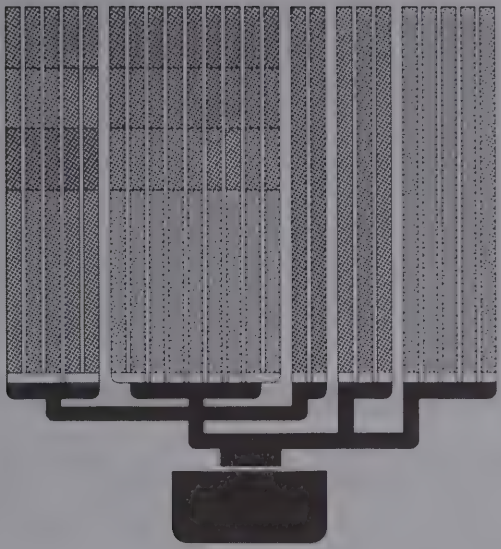
SWITCH SYNCHRONIZER



In many digital systems, a switch is used to initiate a routine or program. This seemingly simple task poses several design problems for the following reasons.

- The switch is activated asynchronously
- It bounces
- It may or may not stay activated longer than it takes to complete the routine.

The circuit shown above implements this function using only one dual flip-flop, the 9020, and a single-pole double-throw (SPDT) switch. When the switch is not activated, FF A is set and FF B is reset (A $\overline{B}$). Depressing the switch activates the $\overline{K}$ input of FF A, causing the next clock pulse to reset it ($\overline{A}$ $\overline{B}$). This activates the J input of FF B, causing the subsequent clock pulse to set it ($\overline{A}$ B). The two flip-flops stay in this state regardless of any changes in the switch because the active Low $\overline{K}$ inputs from the switch are inoperative due to a Low level on the other K inputs. When the routine or program ends, the Response input goes High, causing the next clock pulse to set FF A (A B). If the switch has been released, the subsequent clock pulse resets FF B (A $\overline{B}$), returning the system to the idle state. If the switch has not yet been released, the AB state persists until the switch is released and the next clock pulse resets FF B, returning the system to the idle state. In either case, FF A can be used as a synchronous control flip-flop.



[illegible]

Low Power TTL

11
12
13
14
15

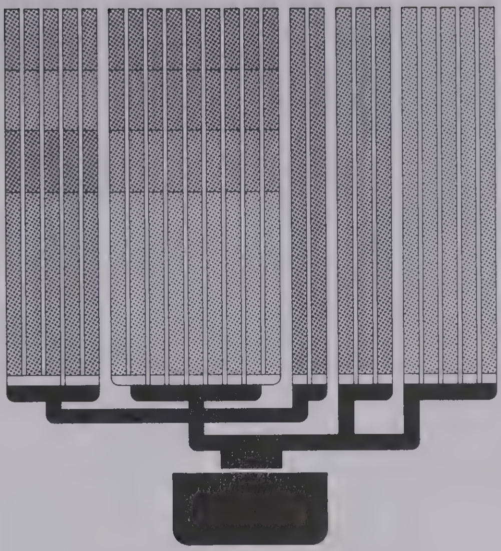
LOW POWER TTL

PRESENTLY AVAILABLE LOW POWER MSI DEVICES

DECODERS/DEMULTIPLEXERS	—93L01	1-of-10 Decoder
	—93L11	1-of-16 Decoder
	—93L21	Dual 1-of-4 Decoder
ENCODERS	—93L18	8-Input Priority Encoder
MULTIPLEXERS	—93L09	Dual 4-Input Multiplexer
	—93L12	8-Input Multiplexer
	—93L22	Quad 2-Input Multiplexer
REGISTERS	—93L00	4-Bit Shift Register
	—93L28	Dual 8-Bit Shift Register
	93L38	Multiport Register
LATCHES	—93L08	Dual 4-Bit Latch
	—93L14	4-Bit Latch
	93L34	8-Bit Latch
OPERATORS	—93L24	5-Bit Comparator
	—93L40	4-Bit Arithmetic Logic Unit
	93L41	4-Bit Arithmetic Logic Unit
COUNTERS	—93L10	Decade Counter
	—93L16	Binary Counter

PRESENTLY AVAILABLE LOW POWER SSI DEVICES

INVERTERS	— 9L04	Hex Inverter (lead identical to 9N04/7104)
GATES	— 9L00	Quad 2-Input NAND (lead identical to 9N00/7400)
	— 9L54	4-Wide, 2-Input AND/OR/INVERT (lead identical to 9N54/7454)
	— 9L86	Quad EXCLUSIVE OR (lead identical to 9N86/7486)
FLIP-FLOPS	— 9L24	Dual J$\overline{K}$ Positive Edge Triggered Flip-Flop (lead identical 9024)



LOW POWER TTL

INTRODUCTION

Over the years, the demands of the computer, instrumentation and communication industries have pushed TTL technology to higher and higher speeds, culminating in the Schottky series of super high speed TTL.

For a large number of applications, however, even standard TTL circuits are faster than required, and it is desirable to trade off the excess speed for a decrease in power consumption. LOW POWER TTL accomplishes this. Fairchild has a comprehensive line of low power TTL/MSI and SSI devices offering many advantages in slow and medium speed systems.

All these circuits are derived from standard TTL/MSI and SSI circuits by effectively increasing all resistors by a factor of four. This reduces all currents, and therefore power consumption, by 75% while the speed is degraded 65%, (i.e., all delays are three times longer).

This gives a new degree of freedom to the systems designer. He can use standard high speed circuits where speed is essential, and can use low power circuits for the sections of the system where speed is not critical.

Fairchild's line of low power TTL/MSI and SSI offers a unique combination of advantages.

LOWER POWER CONSUMPTION

Power consumption is reduced by 75%, resulting in smaller, cheaper power supplies or batteries, reducing the cost, size and weight of the equipment.

LESS HEAT GENERATED

Low power consumption means less heat is generated, simplifying thermal design. Thus, component and board density can increase and fans or ventilation holes may be reduced or even eliminated. Because the low power circuits use the same chip size and package as standard circuits, junction temperatures are lower, increasing reliability.

LESS NOISE GENERATED

Since low power circuits switch only 25% of the current of standard circuits, they generate less noise. Thus, fewer or smaller power supply decoupling capacitors are needed, simplifying printed circuit board design.

LESS NOISE SENSITIVITY

Because it is inherently slower, low power TTL is less susceptible to fast, short duration noise pulses.

ELECTRICAL AND LEAD COMPATIBILITY

The fan out is only reduced by a factor of four, so all low power TTL outputs can drive more than one standard TTL load, and most can drive more than two unit loads. All low power devices are functionally and lead identical with equivalent standard devices. Therefore, a design can easily be changed from standard to low power and vice versa, provided the loading rules are observed.

INCREASED FAN OUT

The effective fan out of standard TTL driving low power TTL is increased by a factor of four. Thus, a standard gate can act as a clock driver for 20 93L00 shift registers and a 9009 buffer can drive as many as 120 low power inputs.

IDEAL FOR MOS TO TTL INTERFACE

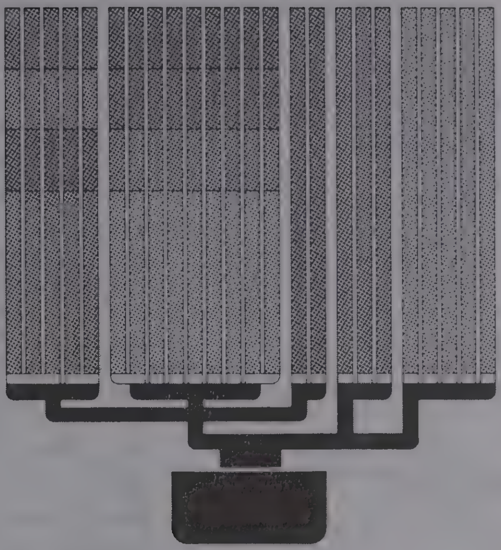
The fan out from MOS circuits is usually quite limited, since MOS is inherently a high impedance process. Most modern MOS circuits provide a fan out of one standard TTL load, which means they can drive four low power TTL inputs.

LONGER DELAYS CAN BE USEFUL

Properly mixing fast and slow circuits can eliminate clock skew, glitch, and race problems. For example, a fast synchronous counter driving a slow low power decoder is less likely to produce glitches, and a properly designed mix of standard and low power shift registers can take advantage of the longer set-up and propagation times and can eliminate clock skew and clock distribution problems.

LARGE NUMBER OF MSI FUNCTIONS

The large number of MSI functions available in low power TTL makes it possible to design entire systems with low power TTL, drastically reducing power consumption.



Schottky Super High Speed TTL

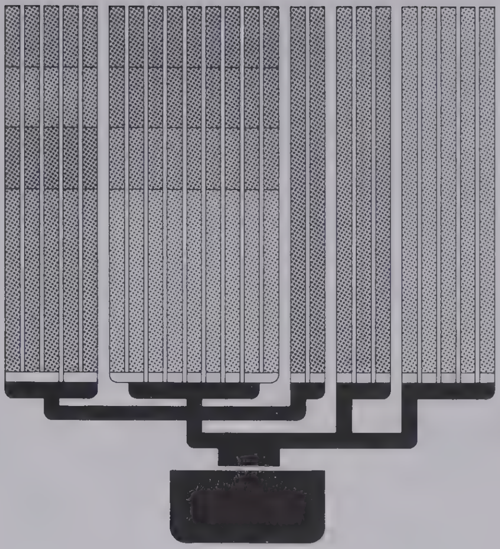
SCHOTTKY TTL

PRESENTLY AVAILABLE SCHOTTKY TTL/MSI

MULTIPLEXERS	93S12	8-Input Multiplexer
	93S13	8-Input Multiplexer with 3-State Output
	93S153,54/74S153	Dual 4-Input Multiplexer
	93S157,54/74S157	Quad 2-Input Multiplexer
	93S158,54/74S158	Quad 2-Input Multiplexer
	93S253,54/74S253	Dual 4-Input Multiplexer with 3-State Outputs
	93S257,54/74S257	Quad 2-Input Multiplexer with 3-State Outputs
	93S258,54/74S258	Quad 2-Input Multiplexer with 3-State Outputs
DECODERS/DEMULTIPLEXERS	93S137	1-of-8 Decoder with Input Latches
	93S138,54/74S138	1-of-8 Decoder/Demultiplexer
	93S139,54/74S139	Dual 1-of-4 Decoder/Demultiplexer
OPERATORS	93S41,54/74S181	4-Bit Arithmetic Logic Unit
	93S42,54/74S182	Carry Lookahead
	93S43,25S05	Twos Complement Multiplier
	93S46	High Speed 6-Bit Comparator
	93S47,71S60/81S60	High Speed 6-Bit Comparator (O.C.)
	93S62,82S62	9-Bit Parity Generator/Checker
REGISTERS	93S00,54/74S195	4-Bit Universal Shift Register
	93S39	8-Bit Multiport Register (to be announced)
	93S174,54/74S174	Hex D Flip-Flop/Register
	93S175,54/74S175	Quad D Flip-Flop/Register
	93S194,54/74S194	4-Bit Bidirectional Shift Register
COUNTERS	93S05	High Speed Variable Modulo Counter
	93S10	Edge Triggered, Fully Synchronous Decade Counter
	93S16	Edge Triggered, Fully Synchronous 4-Bit Binary Counter
MEMORIES	93S404	16-Word x 4-Bit Read/Write Memory (to be announced)

PRESENTLY AVAILABLE SCHOTTKY SSI

NAND GATES/HEX INVERTERS	9S00,54/74S00	Quad 2-Input NAND Gate
	9S03,54/74S03	Quad 2-Input NAND Gate (O.C.)
	9S04,54/74S04	Hex Inverter
	9S04A	High Speed Hex Inverter
	9S05,54/74S05	Hex Inverter (O.C.)
	9S05A	High Speed Hex Inverter (O.C.)
	9S10,54/74S10	Triple 3-Input NAND Gate
	9S20,54/74S20	Dual 4-Input NAND Gate
	9S22,54/74S22	Dual 4-Input NAND Gate (O.C.)
	9S133,54/74S133	13-Input NAND Gate
	9S134,54/74S134	12-Input NAND Gate (3-State Output)
AND GATES	9S11,54/74S11	Triple 3-Input AND Gate
	9S15,54/74S15	Triple 3-Input AND Gate (O.C.)
	9S41	Quad 2-2-3-3 Input AND Gate
NAND BUFFER/DRIVER	9S40,54/74S40	Dual 4-Input Buffer
	9S140,54/74S140	Dual 4-Input Line Driver
AND/OR/INVERT GATES	9S64,54/74S64	4-2-3-2 Input A/O/I Gate
	9S65,54/74S65	4-2-3-2 Input A/O/I Gate (O.C.)
AND/OR GATE	9S42	Dual 4-2 Input A/O Gate
DUAL D FLIP-FLOP	9S74,54/74S74	Positive Edge Triggered D Flip-Flop
DUAL JK FLIP-FLOPS	9S109,54/74S109	Positive Edge Triggered $J\overline{K}$ Flip-Flop
	9S112,54/74S112	Negative Edge Triggered JK Flip-Flop
	9S113,54/74S113	Negative Edge Triggered JK Flip-Flop
	9S114,54/74S114	Negative Edge Triggered JK Flip-Flop



SCHOTTKY TTL

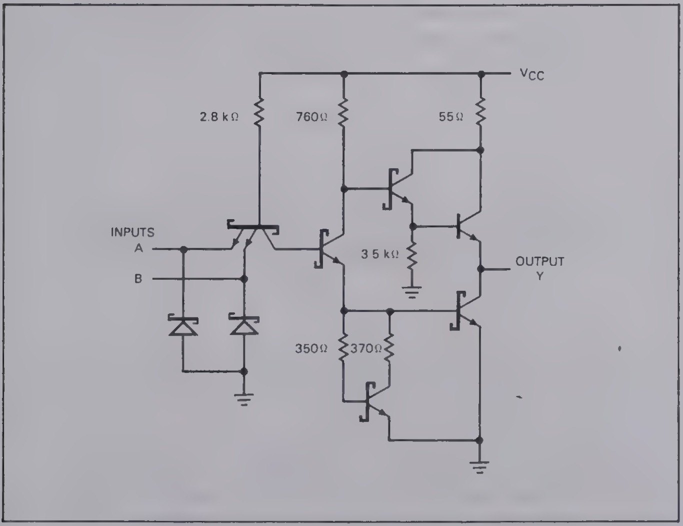
INTRODUCTION

Modern TTL systems can be operated at clock rates over 20 MHz. For considerably faster systems the obvious choice is Emitter Coupled Logic (ECL) which can be operated at system clock rates in excess of 100 MHz, and count up to 500 MHz. But, ECL uses different supply voltages (–5.2 V, –2 V), has less absolute signal swing and noise margins and implements logic in a manner unfamiliar to many TTL designers.

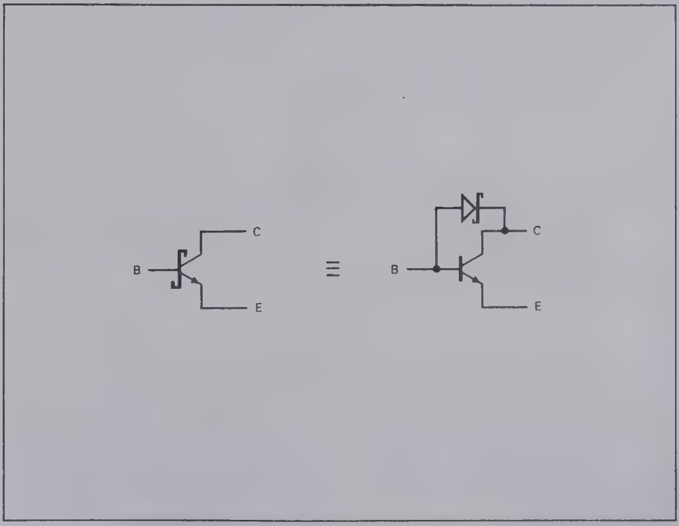
In a system not requiring the speed of ECL, but more speed than conventional TTL offers, Super High Speed (Schottky) TTL is an excellent choice. Schottky TTL circuits, logic configurations, supply voltages and logic levels are identical or very similar to conventional TTL, but Schottky circuits avoid the saturation delay inherent in saturated logic. In such logic (RTL, DTL, TTL) transistors are turned on by applying sufficient base current for the lowest expected current gain. The average transistor, therefore, receives far more base current than necessary, which forward biases the collector base junction and saturates the transistor. In order to turn off such a saturated transistor, the excess base charge must first be removed, resulting in considerable delay. Gold doping is commonly used to speed up the charge recombination, but this decreases the current gain.

Schottky transistors overcome this limitation. They use a surface barrier diode with very low forward voltage drop (0.3 V) and without minority carrier charge storage as a bypass between base and collector. When the transistor starts conducting and is about to become saturated, the excess input current is not fed into the base, but routed through the Schottky diode into the collector (Baker Clamp). As a result, the transistor is never fully saturated and recovers fast when the base current is interrupted. Since gold doping is not required, the transistors also have higher current gain, require less base current, and turn on faster.

2-INPUT NAND GATE



SCHOTTKY TRANSISTOR



SCHOTTKY TTL

ADVANTAGES OF SCHOTTKY TTL

- Through delay is significantly reduced to about half of the delay for High Speed TTL and one third of the delay for standard TTL. This means shorter delays through combinatorial circuits, shorter propagation delays and set-up times for sequential circuits and faster possible clock rates. A Schottky device used as a memory address register is one of the most inexpensive and attractive methods of improving the performance of a high speed memory system.
- Even though the static power consumption of Schottky circuits is generally higher than conventional TTL, it does not increase as rapidly when the circuit is operating at high clock rates. In a fast system, the actual power consumption (static and dynamic) of a Schottky device may be less than its equivalent conventional TTL device.
- Schottky input clamp diodes (0.3 V) are more efficient and suppress ringing better than the clamping diodes of conventional TTL (0.6 V).
- Most Schottky TTL devices are logic, electrical and lead equivalents to conventional TTL devices. This makes them ideal for improving existing systems because interface elements, new supply voltages and basic design changes are not required.

DISADVANTAGES OF SCHOTTKY TTL

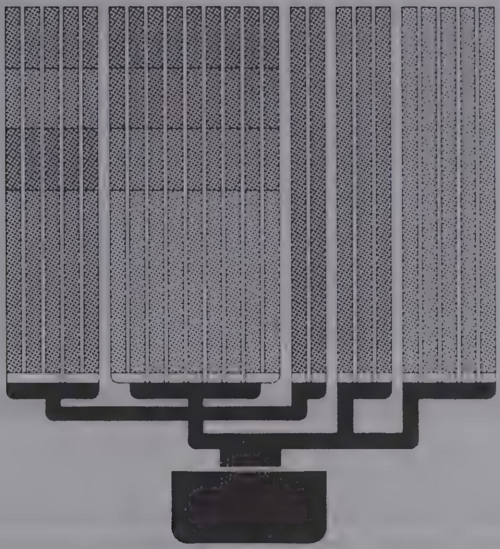
In addition to very attractive advantages, Schottky TTL has some significant disadvantages over conventional TTL.

- Schottky TTL achieves high performance not only by eliminating saturation delays (which is always desirable) but also with considerably faster output transitions. The edge rate of Schottky TTL outputs is typically >1 V/ns (>1000 V/μs), or two to three times steeper than conventional TTL and three to five times steeper than the popular ECL circuits of the 10K and 95K families. This high edge rate causes ringing and other transmission line effects on even moderately long (5-10 inches) PC board interconnections.
- It is difficult to terminate Schottky interconnections properly without wasting fan out capability and power.
- Because the Schottky output transistor does not saturate, $V_{OL\ (max)}$ is 0.5 V and noise immunity in the Low state is 100 mV less than conventional TTL. Instead of 400 mV worst case it is 300 mV worst case when Schottky TTL drives conventional High Speed or Standard TTL, and 200 mV worst case in the exceptional situation when Schottky TTL drives Low Power TTL.

GENERAL RECOMMENDATIONS

High edge rate and reduced noise margins demand very careful PC board layout and much attention to interconnection length, ground and V_{CC} distribution, decoupling, cross talk, and clock distribution problems. It is wise to follow these general recommendations.

- Keep interconnections short
- Use wide, low inductance ground lines or better yet, dedicate one side of a double sided PC board to ground and V_{CC} (ground plane).
- Use generous, distributed supply decoupling with RF type capacitors
- Apply series or parallel termination of long interconnections, but watch the resulting loss in fan out.



Monostable Multivibrators

Triggering Conditions
Non-Retriggerable Operation
Timing Equations and Nomographs
Typical Characteristics
Use of Electrolytic Capacitors
Device Schematics
Operation Descriptions
Pulse Duration Modulator
Pulse Duration Detector
Frequency Discriminator
Schmitt Trigger
Malfunction Indicator
Gated Clock Generator
Unstoppable Oscillator
Voltage Controlled Monostable
Contact Bounce Eliminator
Dual Edge Triggering
Delayed Pulse Generation
Keyed and Synchronized Oscillators

MONOSTABLE SELECTION GUIDE

*Not recommended for new designs, use 9600, see comments below.
**Not recommended for new designs, use 9602, see comments below.

Function	Device	Typical Pulse Width Variations (%) For Commercial Grade Devices		Number Of Inputs		Resettable	Minimum Output Pulse Width, ns (typ)	Power Dissipation, mW (typ)
		0°C — 70°C	4.75 — 5.25 V	Positive	Negative			
Single Retriggerable	9600	±1.5	±1.5	3	2	X	75	125
	9601	±2.7	±1.0	2	2		50	125
Dual Retriggerable	9602	±1.5	±1.5	1	1	X	72	250
	96L02	±0.4	±1.5	1	1	X	110	50
Single Non-Retriggerable	9603,74121	±0.2	±0.15	1	2		40	90
Single Retriggerable	74122*	±2.7	±1.0	2	2	X	45	115
Dual Retriggerable	74123**	±2.7	±1.0	1	1	X	45	230

9601 The original retriggerable monostable.

Advantages — Independent of duty cycle up to 100%. Retriggerable or non-retriggerable operation.

Disadvantages — Temperature coefficient ≈ 10^{-3}/°C. No Schmitt triggered input.

9600 Reset = Clear = Terminate input added to the 9601, outputs come from a latch (reset overrides).

Advantages — Temperature coefficient better than 10^{-3}/°C.

9602 Two 9600s in a package, but with fewer trigger inputs.

Advantages — OR inputs allow non-retriggerable as well as retriggerable operation on either edge. Fewer parts.

Disadvantages — Fewer trigger inputs.

96L02 Low Power version of the 9602. When Speed Permits, Always Use the 96L02!

Advantages — Less power, smaller timing capacitor required. Wider resistor range, (5 kΩ to 220 kΩ). Excellent temperature stability ($0.15 \bullet 10^{-3}$/°C).

Disadvantages — Fewer trigger inputs.

9603/74121 The simplest monostable.

Advantages — Excellent temperature stability ($< 0.1 \bullet 10^{-3}$/°C. One input has hysteresis (Schmitt trigger).

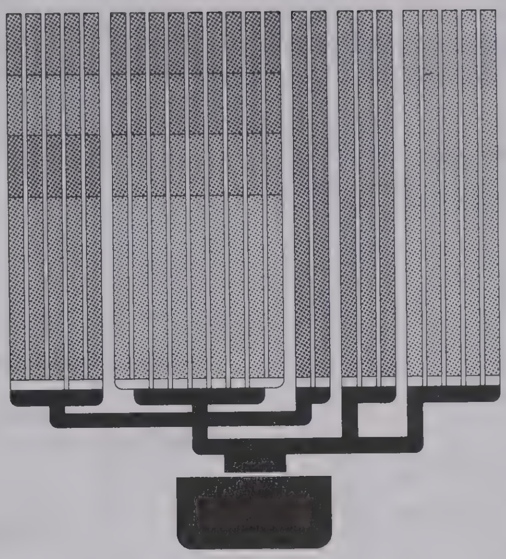
Disadvantages — Non-retriggerable; duty cycle sensitive. Do not use when duty cycle exceeds 60 – 80%.

74122 A poor imitation of the 9600.

Disadvantages — Reset input tied internally to one of the AND trigger inputs. Removing Reset causes the 74122 to trigger if input logic is True $(\overline{1} + \overline{2}) \bullet 3 \bullet 4$.

74123 A poor imitation of the 9602.

Disadvantages — Triggered by AND gate, thus cannot be connected in the non-retriggerable mode without external gates. Chip layout and lead assignment cause crosstalk which is very difficult to overcome with external decoupling. Reset input tied internally to one of the AND trigger inputs. Removing Reset causes the 74123 to trigger if input logic is True $(\overline{1} + 2)$.



MONOSTABLE MULTIVIBRATORS

INTRODUCTION

A monostable multivibrator is a half analog/half digital circuit that produces a pulse on its output in response to a trigger signal at its input. The output pulse width is determined by a resistor/capacitor network.

The 9600 series integrated circuit monostable multivibrators (or one-shots) were designed for ease of application as general purpose delay elements. Without these one-shot circuits, a discrete component one-shot must be built each time a delay element is needed. Also, the 9600 series offers several advantages over the discrete component and earlier integrated circuit one-shots, i.e., DTL 941, DTL 951:

- Retriggerable — the devices can be restarted during a timing cycle (9600, 9601, 9602, 96L02, 74122 and 74123).
- Resettable — a timing cycle can be stopped while in progress and/or inhibited at its start (9600, 9602, 96L02, 74122 and 74123).
- DTL/TTL Compatible — edge triggered inputs provide reliable operation with normal TTL input signal levels. Also, Schmitt trigger inputs provide reliable triggering on waveforms as slow as 1 V/s (9603/74121).
- Operation from a single +5 V supply.
- Large timing pulse width range.

On the other hand, because these monostables are half analog and half digital, they are inherently more sensitive to noise on the analog portion (timing leads) than standard digital circuits. They should not be located near noise-producing sources or transient-carrying conductors and liberal power supply by-passing is recommended for greater reliability and repeatibility. Also, a monostable should not be used as a fix for asynchronous systems; synchronous design techniques always provide better performance.

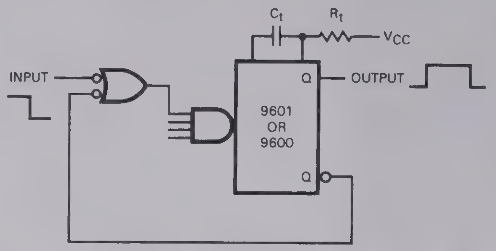
TRIGGERING CONDITIONS

Each of the monostable multivibrators in the 9600 family has multiple inputs to allow triggering of the device on either the Low-to-High or High-to-Low transition of the input waveform. The devices trigger whenever the value of the input logic equation goes from a False to a True condition. To retrigger the 9600, 9601 or 9602, the input logic equation must go from a False to a True condition again (the 9603/74121 is not retriggerable). A continuous True state of the input logic equation does *not* retrigger the device and time out occurs. Only the False-to-True transition triggers or retriggers the devices.

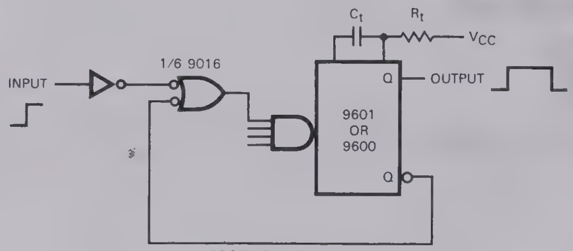
The Boolean expressions for triggering the devices, truth tables and logic symbols appear in the "Description" sections for individual devices.

MONOSTABLE MULTIVIBRATORS

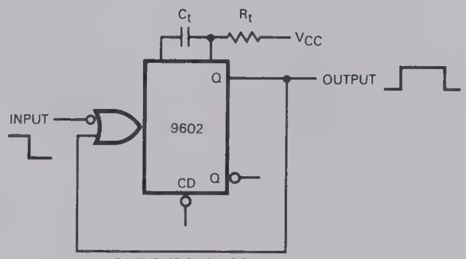
NON-RETRIGGERABLE OPERATION



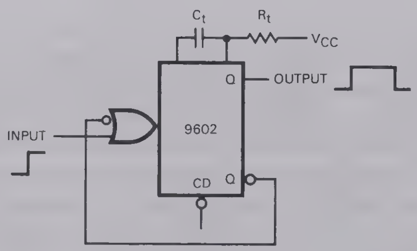
NEGATIVE EDGE TRIGGER



POSITIVE EDGE TRIGGER



NEGATIVE EDGE TRIGGER



POSITIVE EDGE TRIGGER

Note: Unused active level High inputs should be tied High. A floating (unconnected) input is technically in a High logic state, but noise immunity is greatly impaired.

When non-retriggerable operation is required, i.e., when input triggers are ignored during the output cycle, input gating is used to inhibit retriggerability. This configuration may be used for frequency division of a fixed frequency input. Selecting appropriate values of R_t and C_t causes the one-shot to retrigger on some multiple of the input pulses. Note that the 9600 and 9601 require an inverter for non-retriggerable operation using Low-to-High transitions as triggers. The 9602 input gating allows non-retriggerable operation using either positive or negative triggers with no external inverter required. The 9603/74121 is a non-retriggerable monostable, and requires no external wiring or inverter to perform this function.

TIMING EQUATIONS AND NOMOGRAPHS

The timing equation for the 9600, 9601 and 9602 monostable multivibrators is:

$$t = A_1 R_t C_t \left(1 + \frac{A_2}{R_t}\right)$$

where R_t and C_t are the timing resistance and timing capacitance and A_1 and A_2 are multiplication constants.

The timing equation represents the sum of the charge cycle time and the discharge cycle time.

$$t_{charge} = A_1 R_t C_t$$

$$t_{discharge} = A_1 R_t C_t \left(\frac{A_2}{R_t}\right) = A_1 A_2 C_t$$

The A_1 multiplication constant was derived by circuit analysis and verified empirically. For the 9600 and 9601, $A_1 = 0.32$, for the 9602 $A_1 = 0.31$.

The A_2 constant was similarly determined and verified to be 0.7 kΩ for the 9600, 9601 and 1.0 kΩ for the 9602. Thus the timing equations for the 9600, 9601, and 9602 are:

9600: $t = 0.32\, R_t C_t \left(1 + \frac{0.7\text{ k}\Omega}{R_t}\right)$

9601: $t = 0.32\, R_t C_t \left(1 + \frac{0.7\text{ k}\Omega}{R_t}\right)$

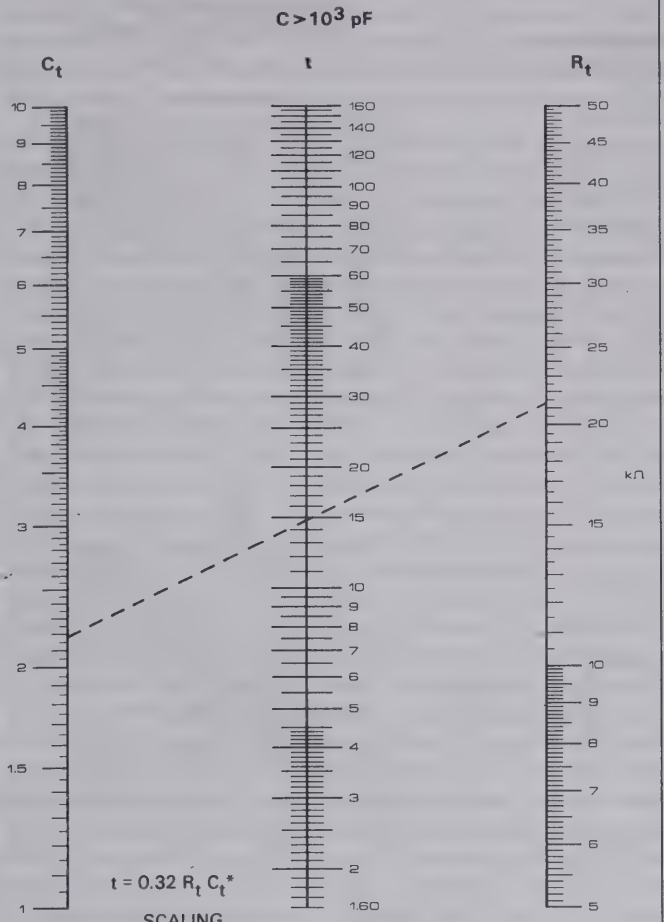
9602: $t = 0.31\, R_t C_t \left(1 + \frac{1.0\text{ k}\Omega}{R_t}\right)$

The above equations apply only to capacitor values greater than 1000 pF. For timing circuits using less than 1000 pF, see Typical Pulse Characteristics curves to determine output pulse width. The above timing equations have a nominal prediction accuracy of ±10%. Thus for precise pulse width applications, trimmer resistors and/or trimmer capacitors should be used to allow setting of the output pulse width to the desired value.

The equation $t = 0.32\, R_t C_t$ may be solved on a nomograph. A straightedge intersecting all three scales is used to solve for either t, R_t or C_t when the other two variables are known. The nomograph does not include the discharge cycle time and therefore is not as accurate as the timing equations. The graphical solution is a good approximation of the values needed to obtain a given pulse width.

MONOSTABLE MULTIVIBRATORS

9600, 9601, 9602 TIMING NOMOGRAPH



$t = 0.32\ R_t\ C_t$*

SCALING

t	C_t
μs	pF x 10^3
μs x 10	pF x 10^4
μs x 100	pF x 10^5
ms	μF
ms x 10	μF x 10
ms x 100	μF x 100
s	μF x 10^3

EXAMPLE: Want t = 150 ms
Have C_t = 22 μF
Use R_t = 21.3 kΩ

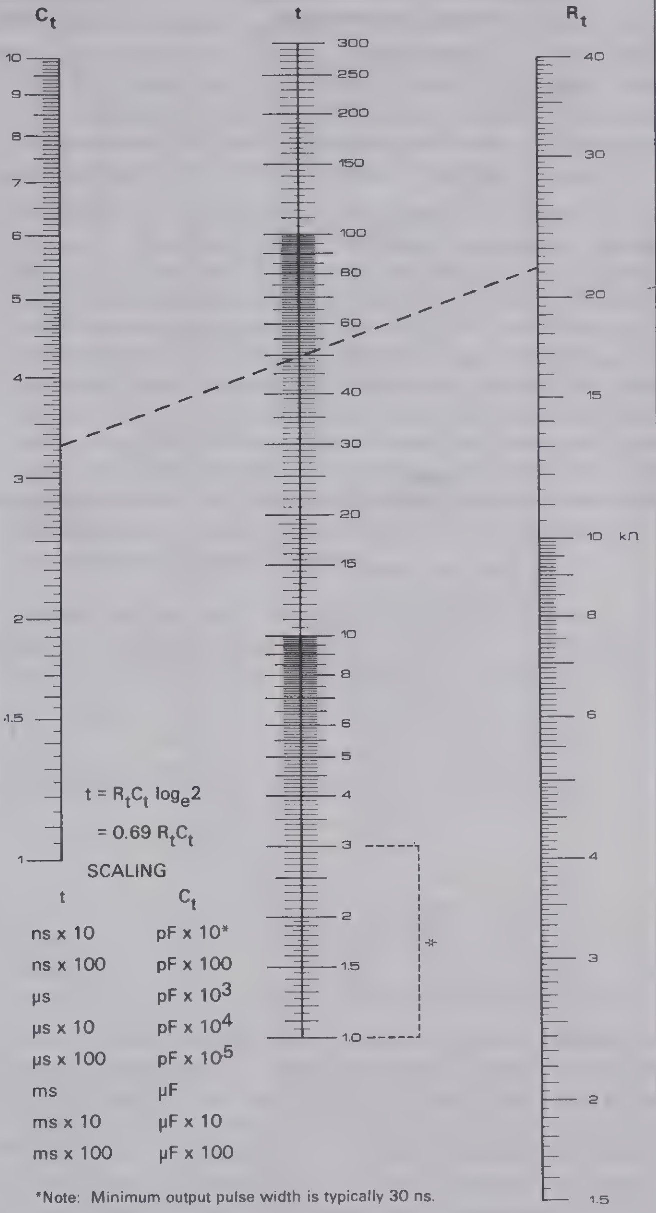
*Approximate equation neglects discharge portion of timing cycle (usually small relative to main or charge cycle time).

The 96L02 is a low power version of the 9602 and has essentially the same circuitry. However, the resistor values in the 96L02 circuit are four times the values of those in the 9602 circuit. The timing equation is:

$$t = 0.33\ R_tC_t\left(1 + \frac{3.0\ \text{k}\Omega}{R_t}\right) \text{ for } C_t > 10^3 \text{ pF}$$

The nominal accuracy of the timing equation is ±10% as long as the capacitor leakages are ≤ 1 μA at 5 V forward bias and ≤ 1.6 μA at 1 V reverse bias. A distinct advantage of the 96L02 is the increased allowable timing resistor range. For the 0 – 75° temperature range device, R_t may be any value from 16 to 220 kΩ.

9603 TIMING NOMOGRAPH



$t = R_tC_t\ \log_e 2$
$= 0.69\ R_tC_t$

SCALING

t	C_t
ns x 10	pF x 10*
ns x 100	pF x 100
μs	pF x 10^3
μs x 10	pF x 10^4
μs x 100	pF x 10^5
ms	μF
ms x 10	μF x 10
ms x 100	μF x 100

*Note: Minimum output pulse width is typically 30 ns.

EXAMPLE: Want t = 50 μs
Have C_t = 3.3 x 10^3 pF
= 0.0033 μF
Need R_t = 21.7 kΩ.

The timing equation for the 9603/74121 consists of the charge cycle time only. Since the 9603/74121 is not retriggerable, the charge cycle time alone determines the output pulse width (The discharge cycle occurs after time out instead of during the timing cycle.) The timing equation is:

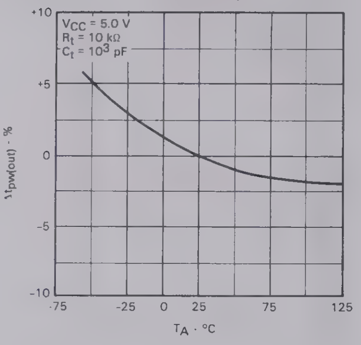
$$t = R_tC_t\log_e 2$$

The timing equation may be solved on the nomograph. A straight line connecting a C_t value and an R_t value intersects the t line so that $t = C_tR_t\log_e 2$. Scale factors relating C_t and t are included.

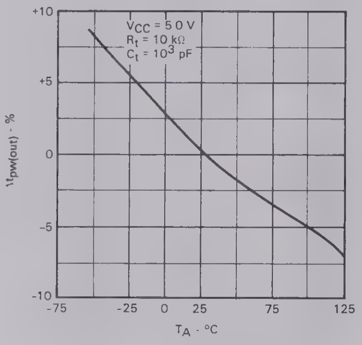
TYPICAL CHARACTERISTICS

Output Pulse Width Variation vs Ambient Temperature

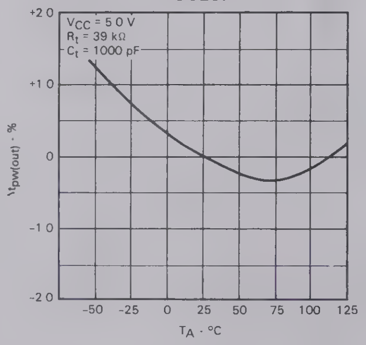
9600/9602



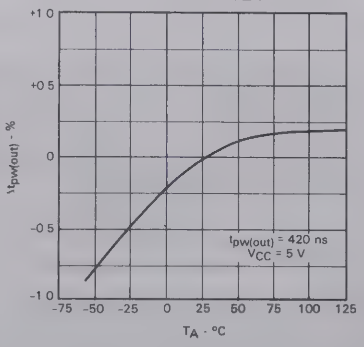
9601



96L02

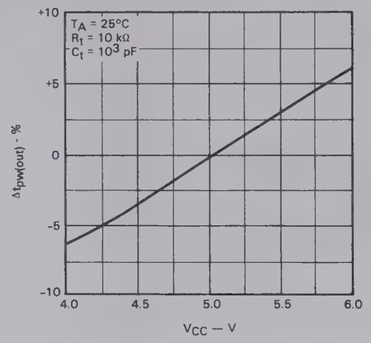


9603/74121

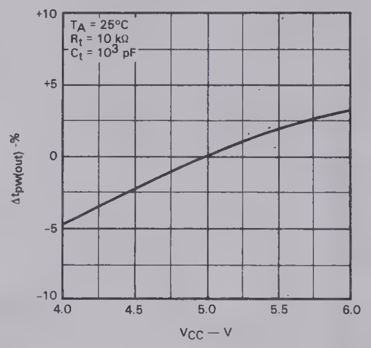


Output Pulse Width Variation vs Power Supply Voltage

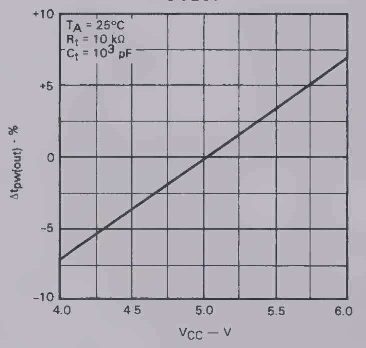
9600/9602



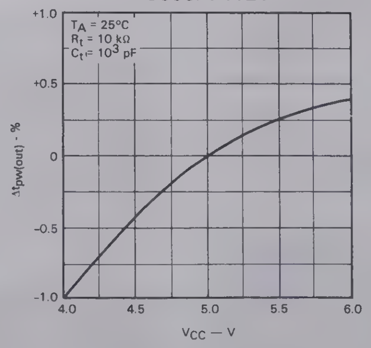
9601



96L02

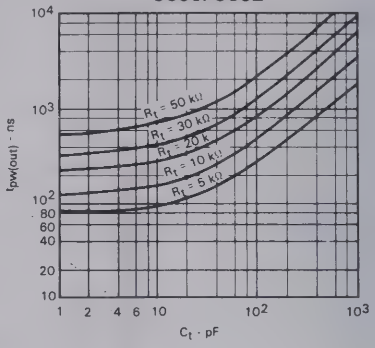


9603/74121

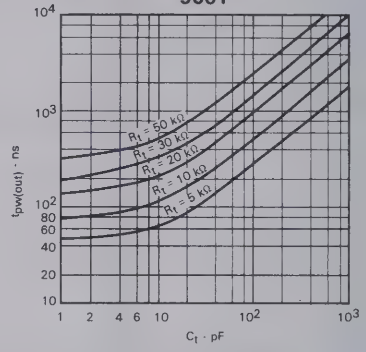


Output Pulse Width vs Timing Capacitance ($C_t < 10^3$ pF)

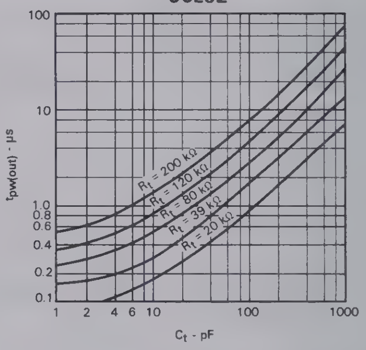
9600/9602



9601

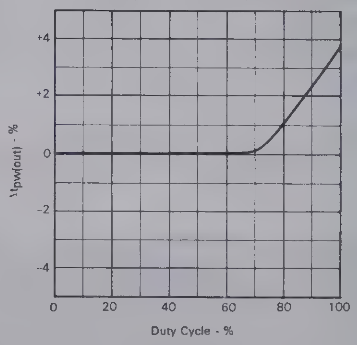


96L02



Output Pulse Width Variation vs % Duty Cycle

9600,9601,9602,96L02

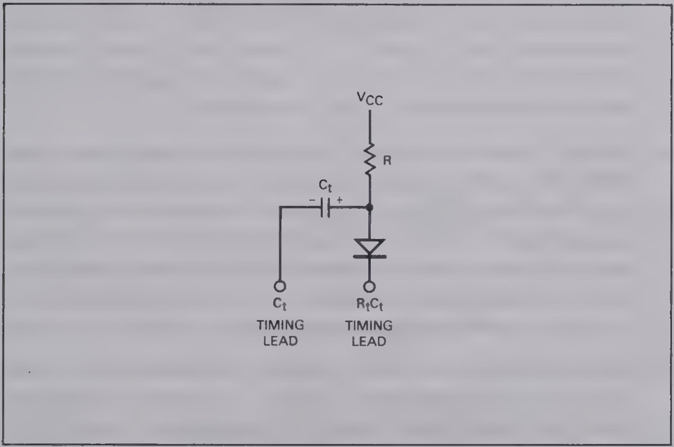


USE OF ELECTROLYTIC TIMING CAPACITORS

When long output pulse widths are required, large value capacitors, usually electrolytic capacitors, are needed. While electrolytic capacitors have the advantage of a high capacitance to volume ratio, they also have the disadvantages of high leakage current and large capacitance tolerances.

As previously indicated, the prediction accuracy of the timing equations for the 9600, 9601, and 9602 is ±10%. This tolerance only accounts for the device-to-device variation, and does not include the effects of timing capacitor leakage current or the value tolerances of the timing resistor or capacitor. (It also excludes V_{CC} and temperature variations from the nominal 5 V V_{CC} at 25°C.) The timing capacitor leakage current has a negligible effect on the timing cycle of the 9600, 9601, 9602 if the leakage is less than 3 μA at 5 V forward bias and less than 5 μA at 1 V reverse bias over the operational temperature range. If the leakage currents are less than the above values, an electrolytic capacitor may be used normally without special circuit modifications; the prediction accuracy of the timing equation (exclusive of timing resistor or capacitor tolerances) remains at ±10%. The 96L02 requires 1/3 the above leakage currents for proper operation, i.e., <1 μA at 5 V forward bias and <1.6 μA at 1 V inverse bias. For the 9601, the differential voltage across the timing capacitor is ±750 mV. Over temperature and V_{CC} the maximum differential voltage can be expected to be less than ±1 V. This requires the capacitor to withstand a repeated −1 V without degradation in the normal R_tC_t configuration for satisfactory one shot operation. Since most tantalum electrolytic capacitors are rated for safe reverse bias operation at up to 5% of their working forward voltage rating, a 20 WVDC capacitor should be used. Capacitor leakage current is usually specified as a constant plus a CV product. This CV product may make the capacitor too "leaky" for good operation if higher working voltage ratings are used. It is helpful to know what working voltage rating provides safe operation at a maximum of 1 V reverse bias. If the capacitor leakage is greater than 3 μA at 5 V forward bias and greater than 5 μA at 1 V reverse bias, one of the following two protective arrangements can be used.

TIMING NETWORK — Diode Protective Arrangement



A diode may be added to the timing network as shown above. While this circuit allows the use of capacitors with greater inverse leakage, it reduces the prediction accuracy of the timing equation and imposes a more restricted timing resistor range.

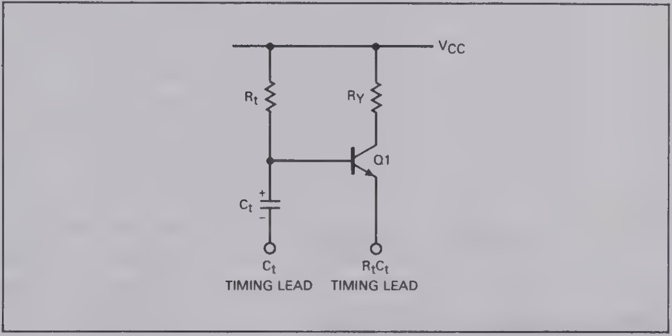
The timing fomula is now:

$$t \cong 0.3\ RC_t \text{ where } R < 0.6\ R_{t(max)}$$

The accuracy of this equation (not including the tolerances of the timing resistor or timing capacitor) is about ±20%. Since the V_f of the diode is uncompensated, the output pulse width variation over the temperature range is increased. Variations in the diode forward conductance will contribute to the degradation of the accuracy of the above equation.

The diode is intended to operate as a fixed offset voltage device to move the differential voltage across the capacitor up from ±1 V to 0 to +2 V. However, the current through the diode varies during the timing cycle and therefore causes the diode forward voltage drop to vary. This can cause the second discharge cycle (in response to a retrigger input) to be less complete than the initial discharge cycle. Since excess charge can be present on the capacitor during the second (retrigger) cycle, this cycle can be much shorter than predicted by the timing equation. The net effect may be early time out on subsequent cycles. The diode configuration is not recommended for applications requiring retriggerable operation, or for use over wide temperature ranges.

TIMING NETWORK — Transistor Capacitance Multiplier Arrangement



In this circuit, a larger timing resistor is used to obtain long output pulse widths. Timing capacitors with high inverse leakage may also be used since the emitter base junction prevents the capacitor from being reverse biased. The transistor Q1 can be any silicon npn transistor with good Beta at low collector currents such as the 2N5961 or 2N5962. The collector resistor R_Y must be in the range of 5 to 50 kΩ (commerical grade) or 5 to 25 kΩ (military grade); the lower values (5 to 10 kΩ) are preferred for improved Beta linearity. The timing resistor value is limited to a range of 5 kΩ to 0.7 $(h_{feQ1})\ R_Y$ or an upper limit of 2 MΩ, whichever is smaller. The timing equation is $t = 0.3\ RC_t$.

The prediction accuracy of the above equation is worse than that for the circuit using the diode. The output pulse width may be more than 30% from the predicted value, depending on the particular transistor and the forward leakage of the timing capacitor.

This circuit also is more temperature sensitive than the normal (R_t and C_t only) circuit. It is not recommended when retriggerable operation is required for the same reasons outlined in the discussion of the diode arrangement. In summary, electrolytic capacitors with low leakage currents are recommended to obtain the highest accuracy and temperature stability where long output pulse widths are required.

SPECIAL CONSIDERATIONS

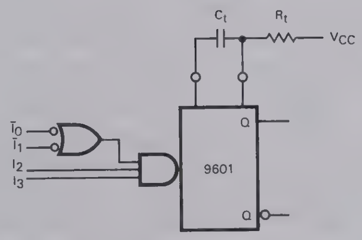
The monostable multivibrator is a practical building block for many applications but it is not a panacea for all system designs. Many times, system performance can be improved by using other types of delay elements. For example, a memory subsystem sequencer can be built with a chain of one shots, but tighter timing tolerances are obtained with a crystal oscillator and a Johnson (twisted ring) shift counter and decoder. For time delays over 1.5 s, or timing capacitors over 100 μF, it is usually better to use a free running astable multivibrator and a couple of inexpensive decade counters (7490) to generate the equivalent of a long delay one shot. Astable oscillators made with monostable building blocks have stabilities approaching 5 parts in 100 and should not be used if system timing is critical. Crystal oscillators provide better stability.

In all one shot applications follow these guidelines:

- Use good high frequency 0.1 μF (ceramic disk) capacitors, located 1 to 2 inches from the monostable package, to bypass V_{CC} to ground.
- Keep timing components (R_t, C_t) close to the package and away from high transient voltage or current carrying conductors.
- Keep the Q output trace away from the "C only" lead; the negative-going edge when the one shot times out may cause the C lead to be pulled down, which may restart the cycle. If this happens, constantly High ($Q = H$, $\overline{Q} = L$) outputs with 50 ns Low spikes will occur at the repetition rate determined by R_t and C_t. If sufficient trace isolation cannot be obtained, a 50 pF capacitor bypassing the C lead to ground usually cures the problem.
- Beware of using the diode or transistor protective arrangement when retriggerable operation is required; the second output pulse may be shorter due to excess charge left on the capacitor. This may result in early time-out and apparent failure of retriggerable operation. Use a good capacitor, one that is able to stand 1 V in reverse and meet the leakage current requirements of the particular one shot.
- Remember that the timing equation associated with each device has a prediction accuracy. Generally, for applications requiring better then ±10% accuracy, trimming to pulse width is necessary.
- Applications involving the 9601 such as the Schmitt trigger and the "unstoppable" astable oscillator will not work with the 9600 or 9602 since the latter have a latch circuit controlling the outputs whereas in the 9601, the internal Schmitt trigger controls the outputs.

9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR

DESCRIPTION

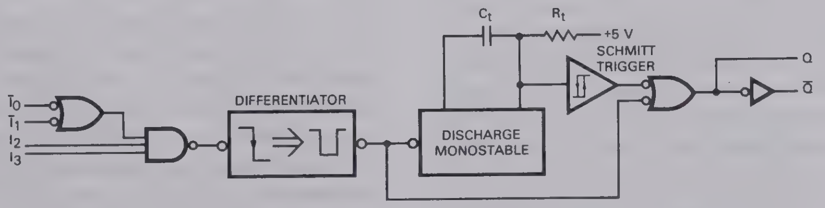


C_t
R_t
V_{CC}
$\overline{I}_0$
$\overline{I}_1$
I_2
I_3
9601
Q
$\overline{Q}$

$T = (\overline{I}_0 + \overline{I}_1) \cdot I_2 \cdot I_3$

$\overline{I}_0$	$\overline{I}_1$	I_2	I_3	OPERATION
H→L	H	H	H	Trigger
H	H→L	H	H	Trigger
L	X	L→H	H	Trigger
X	L	L→H	H	Trigger
L	X	H	L→H	Trigger
X	L	H	L→H	Trigger

INPUTS	LOADING
High	1 UL
Low	1 UL
OUTPUTS	**DRIVE**
High	16 UL
Low	8 UL



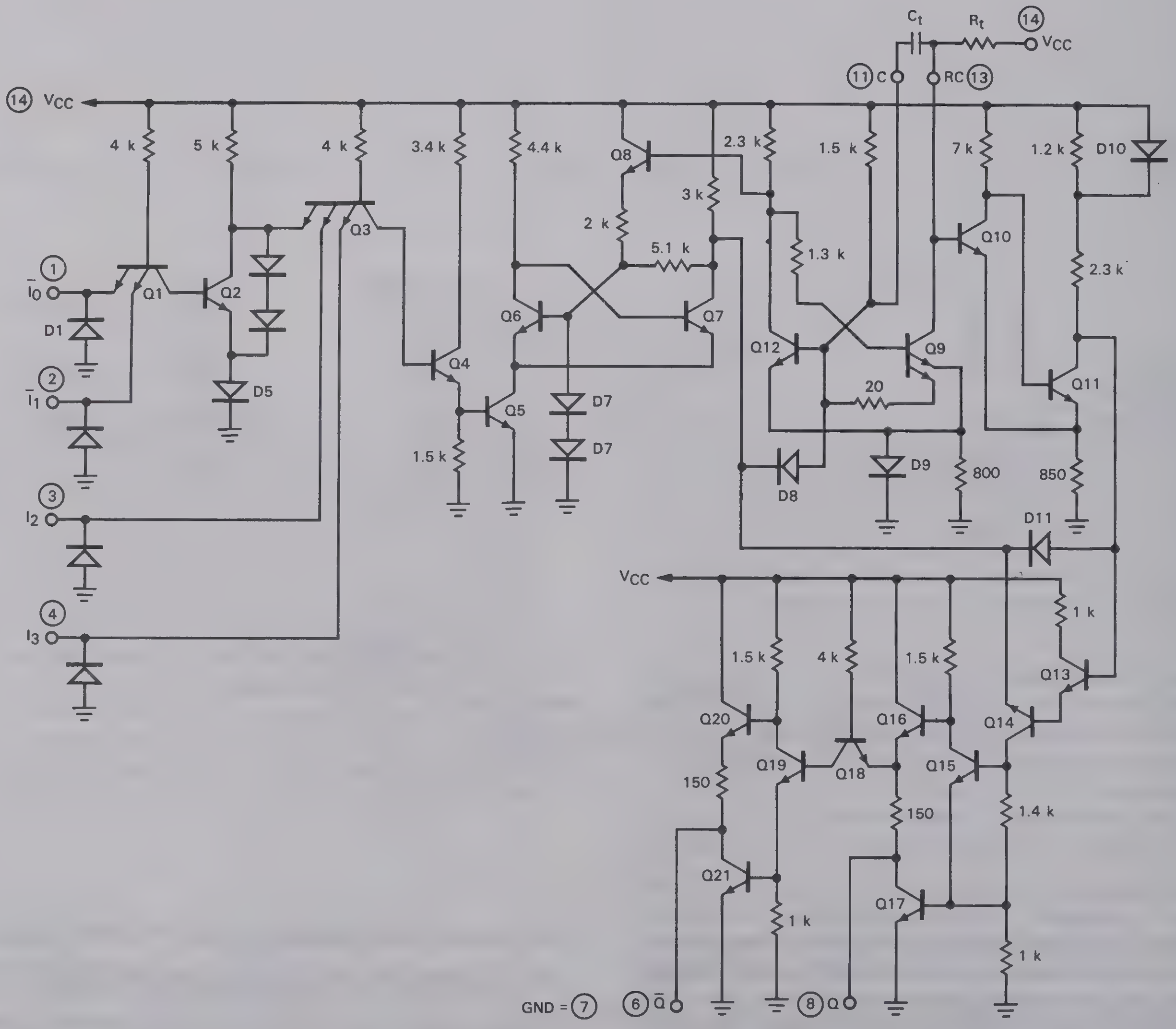
C_t
R_t
+5 V
SCHMITT TRIGGER
Q
$\overline{Q}$
$\overline{I}_0$
$\overline{I}_1$
I_2
I_3
DIFFERENTIATOR
DISCHARGE MONOSTABLE

The 9601 provides an output with duration and accuracy a function of external timing components. It has four inputs, two active High and two active Low, that allow a choice of leading edge or trailing edge triggering. The TTL inputs make triggering independent of input transition times. The 9601 has five subcircuits;

- Input trigger gating logic
- A flip-flop with reset which acts as a differentiator
- A monostable for timing capacitor discharge
- An emitter coupled Schmitt trigger used as a comparator
- An output logic drive circuit to provide both High and Low (Q and $\overline{Q}$) outputs.

9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR

DESCRIPTION



C_t
R_t
⑭ V_{CC}
⑪ C
RC ⑬
⑭ V_{CC}
4 k
5 k
4 k
3.4 k
4.4 k
Q8
2.3 k
1.5 k
7 k
1.2 k
D10
3 k
2 k
5.1 k
1.3 k
Q10
2.3 k
① $\bar{I}_0$
Q1
Q2
Q3
D1
Q6
Q7
Q12
Q9
② $\bar{I}_1$
Q4
D5
Q5
20
Q11
D7
1.5 k
D7
D9
800
850
D8
③ I_2
D11
V_{CC}
1 k
④ I_3
1.5 k
4 k
1.5 k
Q13
Q20
Q16
Q14
Q19
Q18
Q15
150
150
1.4 k
Q21
Q17
1 k
1 k
GND = ⑦
⑥ $\bar{Q}$
⑧ Q

9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR

OPERATION

The five subcircuits of the 9601 are: Input trigger gating logic, Q1 – Q4 plus the trigger transistor Q5; A flip-flop, Q6 and Q7 with reset Q8; A monostable, Q12 and Q9; An emitter coupled Schmitt trigger, Q10 and Q11; An output logic drive circuit, Q13 – Q21.

In the quiescent state, Q5 is off and the discharge monostable is in its stable state (Q12 ON, Q9 OFF). The Schmitt trigger is held on (Q10 ON, Q11 OFF) by the holding current through the timing resistor R_t. With Q11 OFF, Q13 is held on and the Q output is held Low by current through the base collector junction of Q14.

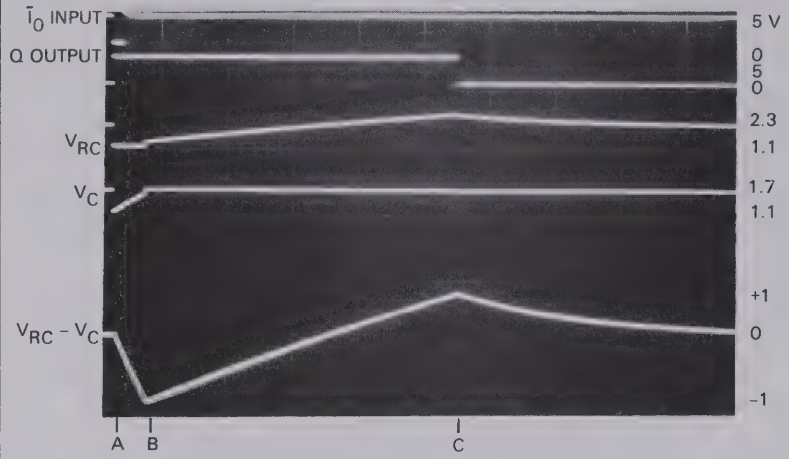
To initiate a timing cycle, Q5 is turned on by the input trigger gating logic Q1 – Q4, and the differentiator flip-flop is set (Q6 OFF, Q7 ON) when Q5 goes into saturation. When Q7 turns on, Q15 is turned off through the emitter of Q14, thus forcing the Q output to go High and the $\overline{Q}$ output to go Low. When Q7 turns on it also sets the discharge monostable (Q12 and Q9) to its unstable state (Q12 OFF, Q9 ON) by turning Q12 off with current through D8. The extra emitter on Q9 allows Q9 to turn on at the same time that Q12 turns off. Turning Q12 off allows Q8 to supply base drive to Q6, turning Q6 on and restoring the differentiator flip-flop to its stable state (Q6 ON, Q7 OFF). Thus, the collector of Q7 produces a narrow, negative going pulse approximately 50 ns wide. Simultaneously, the collector of Q9 pulls the base of Q10 down and resets the Schmitt trigger (Q10 OFF, Q11 ON). Q11 causes Q13 to turn off, removing the base drive to Q15. This insures that Q15 remains off with the Q output High during the remainder of the timing cycle. Q14 provides a speedup path to bypass the time delay through the Schmitt trigger section. When the discharge monostable is set (Q12 OFF, Q9 ON), timing capacitor C_t is allowed to discharge (actually charge in reverse) into the collector of Q9 and D9, with current supplied through the 1.5 kΩ resistor on the base of Q12. After C_t loses sufficient charge, the emitter base junction of Q12 is forward biased, turning Q12 on; the monostable now returns to its stable state (Q12 ON, Q9 OFF). C_t then begins charging toward V_{CC} through R_t. Because the 1.5 kΩ resistance is much less than R_t, discharge time is only a small percent of the total timing cycle. During the charge portion of the timing cycle, C_t charges with the charge path through the base emitter junction of Q12 and D9 and charge current supplied through R_t.

When the timing capacitor is charged to the threshold of the Schmitt trigger, the timing cycle ends as Q10 turns on and Q11 turns off. As Q11 turns off, Q13 turns on, turning Q15 on again through the base collector junction of Q14. As Q15 goes on, the Q and $\overline{Q}$ outputs are restored to their quiscent states: Q is Low and $\overline{Q}$ is High. The Schmitt trigger input impedance changes from high to low when input voltage exceeds threshold voltage. This input impedance remains low as long as the voltage does not drop below the turn off threshold. Thus, the timing capacitor discharges into the base of Q10 and the 860 Ω emitter resistor of Q11. The maximum limit on the value of the timing resistor insures that Q10 receives sufficient base drive to remain on after the timing capacitor has discharged (off state). Because the Schmitt trigger remains fired, the Q and $\overline{Q}$ outputs remain in their quiescent states ($Q = L$, $\overline{Q} = H$). The minimum value for the timing resistor insures that Q9 is not unsaturated during the discharge cycle by excess collector current.

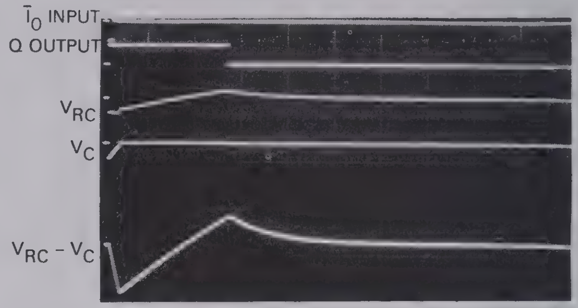
Retriggering of the 9601 occurs when Q5 is turned off and then on unless the discharge monostable is already in the unstable state (Q12 OFF, Q9 ON). Retrigger pulses are ignored during the discharge cycle because the cycle is self-sustaining and only resets when C_t has discharged enough to forward bias the base emitter junction of Q12, allowing the discharge monostable to reset. Retrigger pulses, occuring after the discharge cycle but before the Schmitt trigger turns on to end the timing cycle, cause the discharge cycle to start again but the outputs remain in the High state ($Q = H$, $\overline{Q} = L$).

9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR

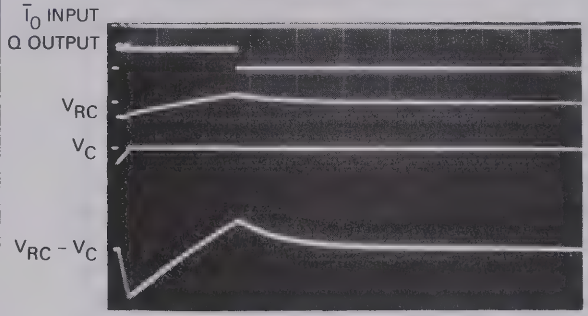
TIMING WAVEFORMS



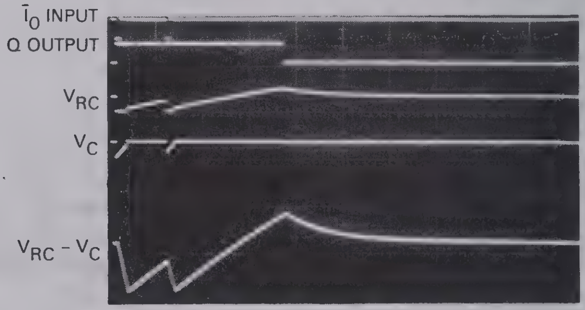
a. **SINGLE TIMING CYCLE**



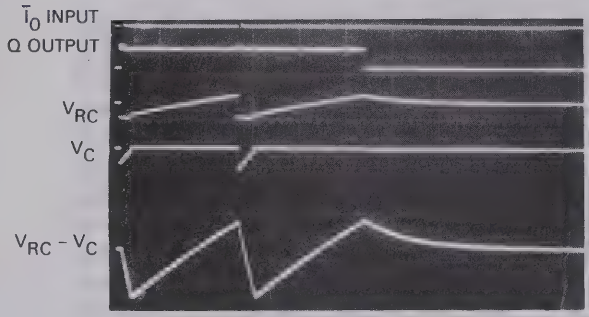
b. TIMING CYCLE (Same as a. with reduced time scale)



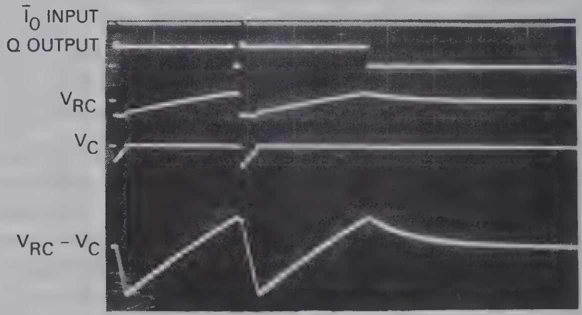
c. **RETRIGGERS DURING DISCHARGE CYCLE ARE IGNORED**



d. **RETRIGGERABLE OPERATION** $V_C < V_C$ **QUIESCENT**



e. **RETRIGGERABLE OPERATION** $V_C > V_C$ **QUIESCENT**



f. **TRIGGER AFTER TIME OUT**

Typical waveforms of a single timing cycle are shown in *a* above. The top waveform is the input trigger (the monostable triggers on the negative edge of the input trigger). The second trace is the Q output. The third trace is the voltage waveform at the RC lead (input to Schmitt trigger), and the fourth trace is the voltage waveform at the C lead (base of Q12). The fifth trace is the differential voltage across the timing capacitor.

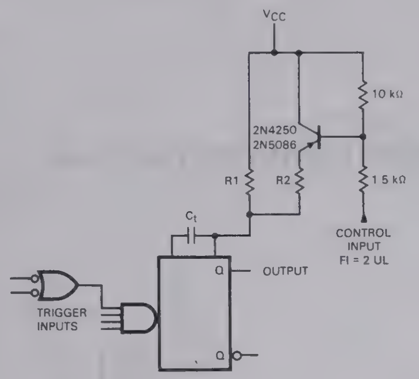
On the negative edge of the trigger input (point A in *a*), trigger transistor Q5 turns on, firing the differentiator flip-flop. The negative going pulse at Q7 causes: the discharge monostable to set; the C lead (base of Q12) to be pulled down (trace 4) by D8; the RC lead to be clamped to $V_{CE(SAT)Q9} + V_{D9}$ and; the Schmitt trigger to reset Q11, causing the Q output to go High and the $\overline{Q}$ output to go Low.

The timing capacitor charges in the reverse direction until the base emitter of Q12 is forward biased (point B in *a*); this causes the discharge monostable to reset (Q12 ON, Q9 OFF). With Q12 on, the C lead is clamped by the base emitter of Q12 and D9, and the timing capacitor is allowed to charge through R_t. When C_t has charged to the threshold voltage, the Schmitt trigger fires (point C in *a*), the Q output goes Low, the $\overline{Q}$ output goes High and the timing cycle is complete.

Traces in *b* are the same as those in *a* but at a reduced sweep speed. A retrigger pulse during the discharge cycle (note that retriggering does not occur) is shown in *c*, while *d* shows a retrigger cycle. In this case, the discharge cycle takes less time than the initial cycle since C_t possesses less charge at retrigger time than it had at the start of the cycle. A trigger pulse occurring just after time out (note the increased discharge time on the second cycle) is shown in *f*. The increased discharge time is responsible for the increase in output pulse width as a function of duty cycle.

9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR APPLICATIONS

PROGRAMMABLE DELAY TIME — PDM GENERATOR



VCC
10 kΩ
2N4250
2N5086
R1
R2
1.5 kΩ
Ct
CONTROL INPUT FI = 2 UL
Q
OUTPUT
TRIGGER INPUTS
Q

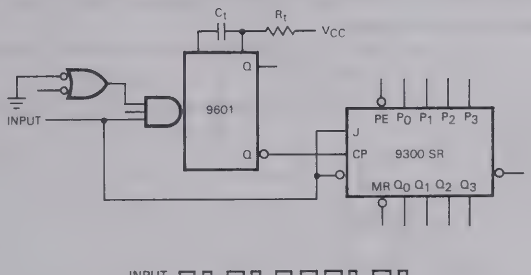
The 9600 family may be programmed to have either of two delay times. The transistor is a high beta pnp (2N4250 or 2N5086) used in the inverse saturation mode as a low offset voltage analog switch. The 10 kΩ and 1.5 kΩ resistors allow programming with TTL logic gates. The fan in is 2 unit loads.

A logic High at the control input turns the transistor off, so that delay time is determined by R1 and C_t. A logic Low at the control input turns the transistor on, reducing the time delay. The lower delay time is determined by C_t and the parallel combination of R1 and R2.

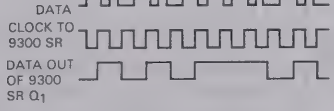
$$R_{effective} = \frac{R1R2}{R1 + R2}$$

For PDM modulator applications, the control input becomes the Data input and a clock is fed to the appropriate trigger input. C_t, R1 and R2 are set so that C_tR1 produces a 66% duty cycle pulse with respect to the clock. The $C_tR_{effective}$ produces a 33% duty cycle pulse. When the control input (Data) is High, a 66% duty cycle pulse is sent; when it is Low, a 33% duty cycle pulse is sent.

PULSE DURATION DETECTOR



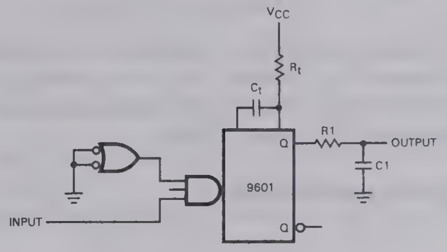
Ct
Rt
VCC
Q
INPUT
9601
Q
PE P0 P1 P2 P3
J
CP
9300 SR
MR Q0 Q1 Q2 Q3



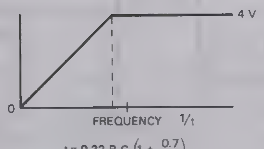
INPUT DATA
CLOCK TO 9300 SR
DATA OUT OF 9300 SR Q1

A circuit for decoding PDM signals is shown above. If the input data presents a logic zero (33% duty cycle) to the 9601 detector, the timing cycle is longer than the input data pulse. The $\overline{Q}$ output of the 9601 returning to its normally High state clocks a logic zero into the 9300 4-bit shift register. An input logic one signal (66% duty cycle) is still High after the 9601 times out, thereby clocking a logic one into the shift register.

FREQUENCY DISCRIMINATOR



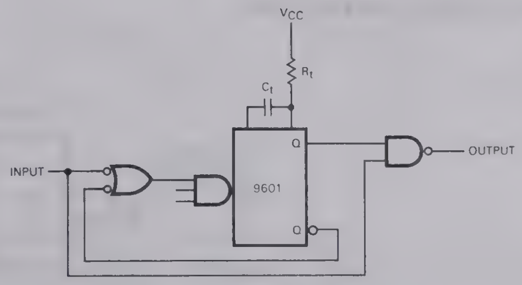
VCC
Rt
Ct
Q
R1
OUTPUT
C1
9601
INPUT
Q



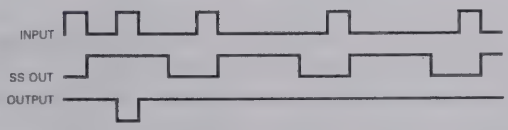
$t = 0.32\,R_tC_t\left(1 + \frac{0.7}{R_t}\right)$

A simple resistor-capacitor integration network may be used on the 9601 output to produce a linear output voltage curve proportional to frequency over a limited range. For a given value of R_t and C_t, the output duty cycle is directly proportional to the input frequency. The duty cycle output is integrated by R1 and C1 to produce the output voltage.

DOUBLE PULSE DETECTION



VCC
Rt
Ct
Q
OUTPUT
INPUT
9601
Q

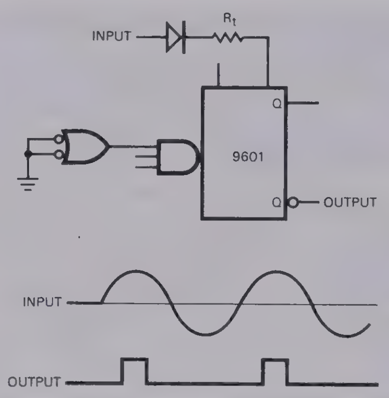


INPUT
SS OUT
OUTPUT

The 9601 may be used to detect an index point on a rotating drum or disk. If a double pulse is recorded to indicate an index point, this circuit will detect the index.

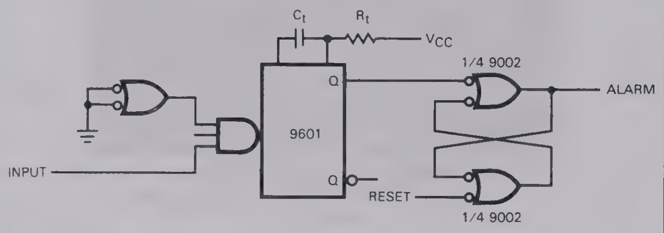
9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR APPLICATIONS

SCHMITT TRIGGER (960l only)


INPUT
R_t
Q
9601
$\overline{Q}$
OUTPUT
INPUT
OUTPUT

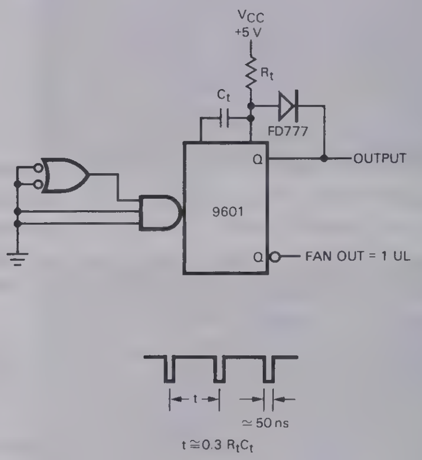
Schmitt trigger operation is provided by a triggering signal through R_t directly into the Schmitt trigger section of the 9601. Input peak amplitude should supply >100 μA at 2.5 V and <1 mA.

MALFUNCTION INDICATOR


C_t
R_t
V_{CC}
1/4 9002
ALARM
Q
9601
INPUT
$\overline{Q}$
RESET
1/4 9002

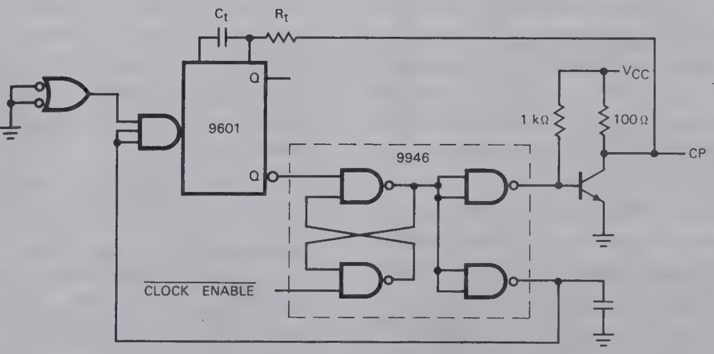
A monitor to detect low speed malfunctions in a system may be constructed using the 9601 plus gating circuitry. For fixed values of R_t and C_t, the 9601 retriggers if the triggering pulse period is less than the timing cycle. The Q output remains High as long as the 9601 continues to retrigger. A missed input pulse or a decrease in the input frequency allows the 9601 to time out, thereby setting the latch and indicating low speed malfunction. Applications include clock pulse and motor speed monitoring.

UNSTOPPABLE OSCILLATOR (9601 only)


V_{CC}
+5 V
R_t
C_t
FD777
Q
OUTPUT
9601
$\overline{Q}$
FAN OUT = 1 UL
t
≃ 50 ns
t ≅ 0.3 R_tC_t

In the 9601 one shot, the Schmitt trigger circuit (connected internally) controls the state of the outputs. If the Schmitt trigger is above its upper threshold, the outputs are Low ($Q = L$, $\overline{Q} = H$); when it is below its lower threshold, the outputs are High ($Q = H$, $\overline{Q} = L$). By connecting a diode, positive feedback is obtained causing the 9601 circuit to become an astable oscillator. If the Q output is Low, the Schmitt trigger turns off, causing a charge cycle (Q goes High). Upon time out, the cycle repeats. This results in a pulse output from Q which has a repetition rate determined by the R_tC_t network. The inputs are grounded since the normal triggering path is not used in this application. A wider pulse can be obtained by connecting the diode to the C lead instead of the RC lead.

GATED CLOCK GENERATOR

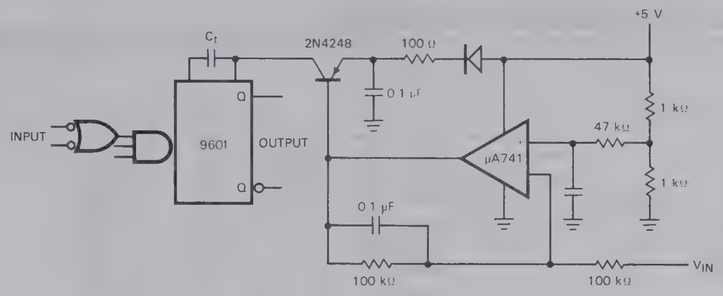

C_t
R_t
Q
9601
$\overline{Q}$
V_{CC}
1 kΩ
100 Ω
CP
9946
$\overline{\text{CLOCK ENABLE}}$

The first clock pulse is generated immediately after applying Clock Enable (active Low). Clock pulse width is unaffected by any change on the Clock Enable input.

Feeding the timing resistor from the Clock output prevents accidental latch up and the output transistor provides a fan out of >50 unit loads.

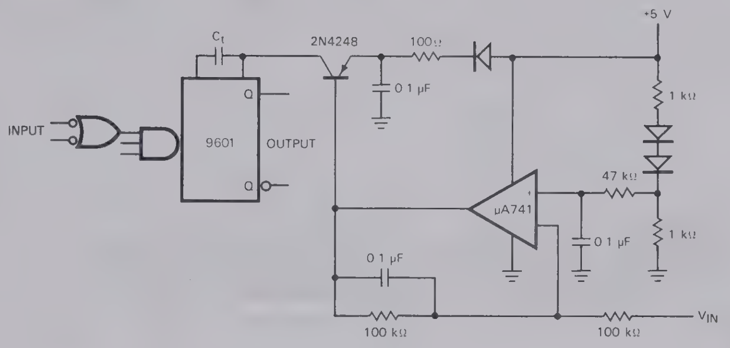
9601 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR APPLICATIONS

VOLTAGE CONTROLLED MONOSTABLE



+5 V
C_t
2N4248
100 Ω
0.1 μF
INPUT
9601
Q
OUTPUT
Q̄
47 kΩ
1 kΩ
1 kΩ
μA741
0.1 μF
100 kΩ
100 kΩ
V_{IN}

a.

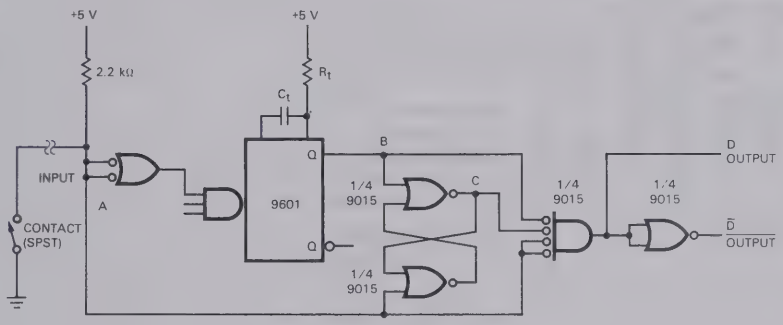


+5 V
C_t
2N4248
100 Ω
0.1 μF
INPUT
9601
Q
OUTPUT
Q̄
1 kΩ
47 kΩ
μA741
0.1 μF
1 kΩ
0.1 μF
100 kΩ
100 kΩ
V_{IN}

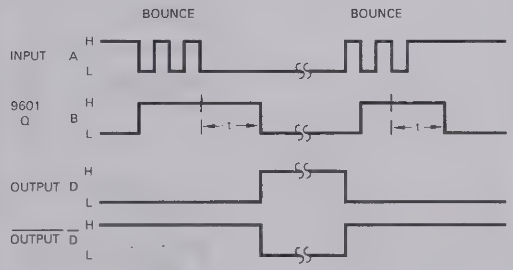
b.

The 9601 may be used as a voltage controlled one shot using a single TTL power supply. Circuit operation requires the use of the 2N4248 as a current source driven by the μA741 to control the output pulse width. When used as shown in *a*, the control range will extend in a positive direction from two diode drops above ground (maximum pulse width). An additional two diodes, placed as shown in *b*, offset the control range to ground and, at the same time, stabilize the operation over temperature. The current control range should not extend lower than 60 μA for the 0 – 75°C range 9601.

FORM "A" (SPST) CONTACT BOUNCE ELIMINATOR



+5 V
2.2 kΩ
+5 V
R_t
C_t
INPUT
A
CONTACT (SPST)
9601
Q
Q̄
B
1/4 9015
C
1/4 9015
1/4 9015
1/4 9015
D OUTPUT
D̄ OUTPUT



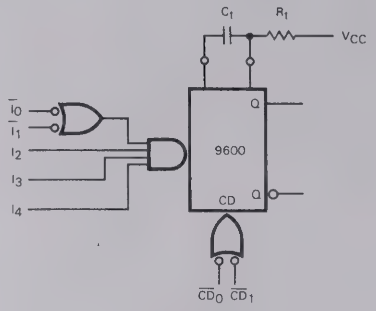
BOUNCE
BOUNCE
INPUT A
H
L
9601 Q B
H
L
t
t
OUTPUT D
H
L
OUTPUT D̄
H
L

The use of a Form C (SPDT) contact and an RS latch formed by cross-coupled gates is a widely accepted method of removing contact bounce on a mechanical to logic interface. In some systems, a Form C contact arrangement is not available, but instead, a Form A (SPST,N.O.) contact must be used. The circuit above will remove the bounce from a Form A contact with only two resistors, one capacitor, and two digital logic packages (one 9601, one 9015).

The 9601 triggers each time the input A goes Low, causing the Q output to go High. The 9601 time delay is set by R_tC_t to be longer than the bounce period (t of 10 ms is usually adequate). Two of the NOR gates in the 9015 are connected as an RS latch to remember that a contact closure has occured. The 4-input NOR in the 9015 is used as a negated input AND gate that provides a High output when the contact has closed AND the 9601 has triggered AND it has timed out ($D = \overline{A} \bullet \overline{B} \bullet \overline{C}$). The logic shown is needed to avoid output glitches due to the difference in propagation delays of the 9601 and the RS latch. The remaining NOR gate in the 9015 is used to provide the complement of point D and is in the same logic sense as the contact closure.

9600/9602/96L02 RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATORS

9600 SINGLE MONOSTABLE MULTIVIBRATOR DESCRIPTION



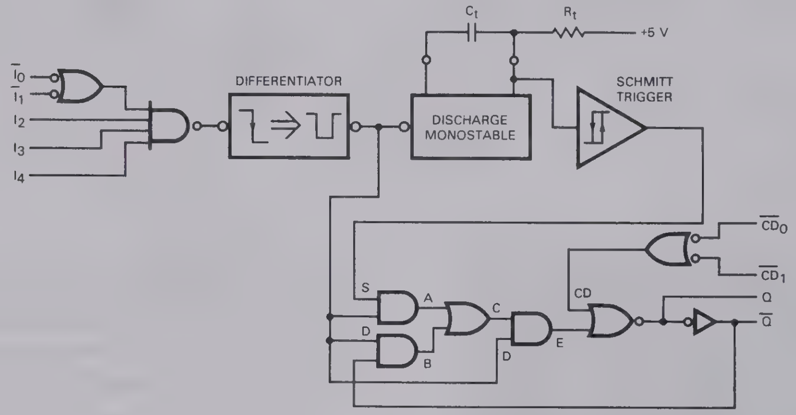
C_t
R_t
V_{CC}
$\overline{I}_0$
$\overline{I}_1$
I_2
I_3
I_4
9600
Q
CD
$\overline{Q}$
$\overline{CD}_0$
$\overline{CD}_1$

$T = (\overline{I}_0 + \overline{I}_1) \bullet I_2 \bullet I_3 \bullet I_4$
($\overline{CD}_0$ and $\overline{CD}_1$ must be H)

$\overline{I}_0$	$\overline{I}_1$	I_2	I_3	I_4	$\overline{CD}_0$	$\overline{CD}_1$	OPERATION
H→L	H	H	H	H	H	H	Trigger
H	H→L	H	H	H	H	H	Trigger
L	X	L→H	H	H	H	H	Trigger
X	L	L→H	H	H	H	H	Trigger
L	X	H	L→H	H	H	H	Trigger
X	L	H	L→H	H	H	H	Trigger
L	X	H	H	L→H	H	H	Trigger
X	L	H	H	L→H	H	H	Trigger
X	X	X	X	X	L	X	Reset
X	X	X	X	X	X	L	Reset

FAN IN INPUTS	FAN OUT LOADING
High	1 UL
Low	1 UL

OUTPUTS	DRIVE
High	16 UL
Low	8 UL



C_t
R_t
+5 V
$\overline{I}_0$
$\overline{I}_1$
I_2
I_3
I_4
DIFFERENTIATOR
DISCHARGE MONOSTABLE
SCHMITT TRIGGER
$\overline{CD}_0$
$\overline{CD}_1$
S
A
C
D
B
D
E
CD
Q
$\overline{Q}$

INTERNAL STATE SEQUENCE
(Note: Delays exaggerated to show sequence of events)
Refer to Internal Logic Diagram

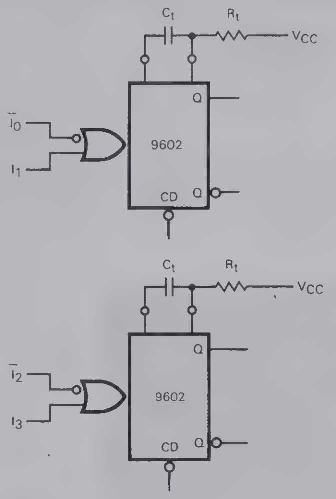
OPERATION	D	S	A	B	C	E	CD	Q	$\overline{Q}$	COMMENT
Quiescent State	H	H	H	H	H	H	L	L	H	
Differentiator Starts	L	H	H	H	H	H	L	L	H	Triggered
	L	H	L	L	H	L	L	L	H	—Start Dis-
	L	H	L	L	L	L	L	H	H	charge Cycle
	L	H	L	L	L	L	L	H	L	
Schmitt Trigger Turns Off	L	L	L	L	L	L	L	H	L	
Differentiator Restores	H	L	L	L	L	L	L	H	L	
										Charge Cycle
Schmitt Trigger Turn On	H	H	L	L	L	L	L	H	L	End of Cycle
	H	H	H	L	L	L	L	H	L	
	H	H	H	L	H	L	L	H	L	
	H	H	H	L	H	H	L	H	L	
	H	H	H	L	H	H	L	L	L	
	H	H	H	L	H	H	L	L	H	
Quiescent State	H	H	H	H	H	H	L	L	H	

The 9600 circuit is similar to the 9601 except for the following.

- Different input gating arrangement
- Improved temperature compensation on Schmitt trigger
- Added gate circuitry for reset inputs and feedback from $\overline{Q}$ output to allow reset operation.

9600/9602/96L02 RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATORS

9602/96L02 DUAL MONOSTABLE MULTIVIBRATOR DESCRIPTION



C_t
R_t
V_{CC}
$\overline{I}_0$
I_1
9602
Q
CD
$\overline{Q}$
$\overline{I}_2$
I_3

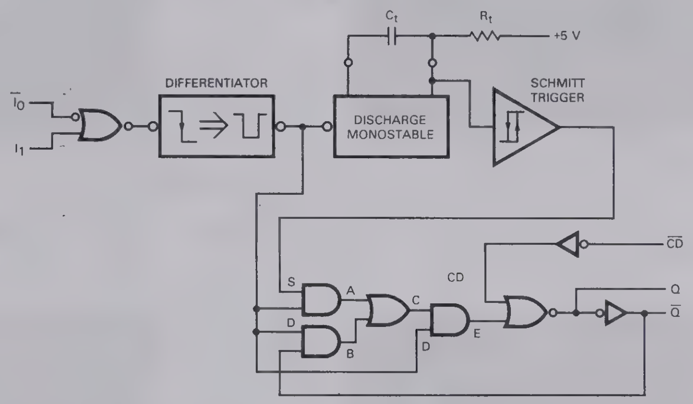
$T_1 = \overline{I}_0 + I_1$ (Monostable 1)
$T_2 = \overline{I}_2 + I_3$ (Monostable 2)
($\overline{CD}$ inputs are H)

I_1 I_3	$\overline{I}_0$ $\overline{I}_2$	$\overline{CD}_0$ $\overline{CD}_1$	OPERATION
H→L	L	H	Trigger
H	L→H	H	Trigger
X	X	L	Reset

INPUTS	9602 LOADING	96L02 LOADING
High	1 UL	0.5 UL
Low	1 UL	0.66 UL

OUTPUTS	DRIVE	DRIVE
High	16 UL	9 UL
Low	8 UL	3 UL

1/2 9602/96L02



C_t
R_t
+5 V
DIFFERENTIATOR
DISCHARGE MONOSTABLE
SCHMITT TRIGGER
$\overline{I}_0$
I_1
$\overline{CD}$
S
A
C
CD
D
E
B
Q
$\overline{Q}$

INTERNAL STATE SEQUENCE

(Note: Delays exaggerated to show sequence of events)

Refer to Internal Logic Diagram

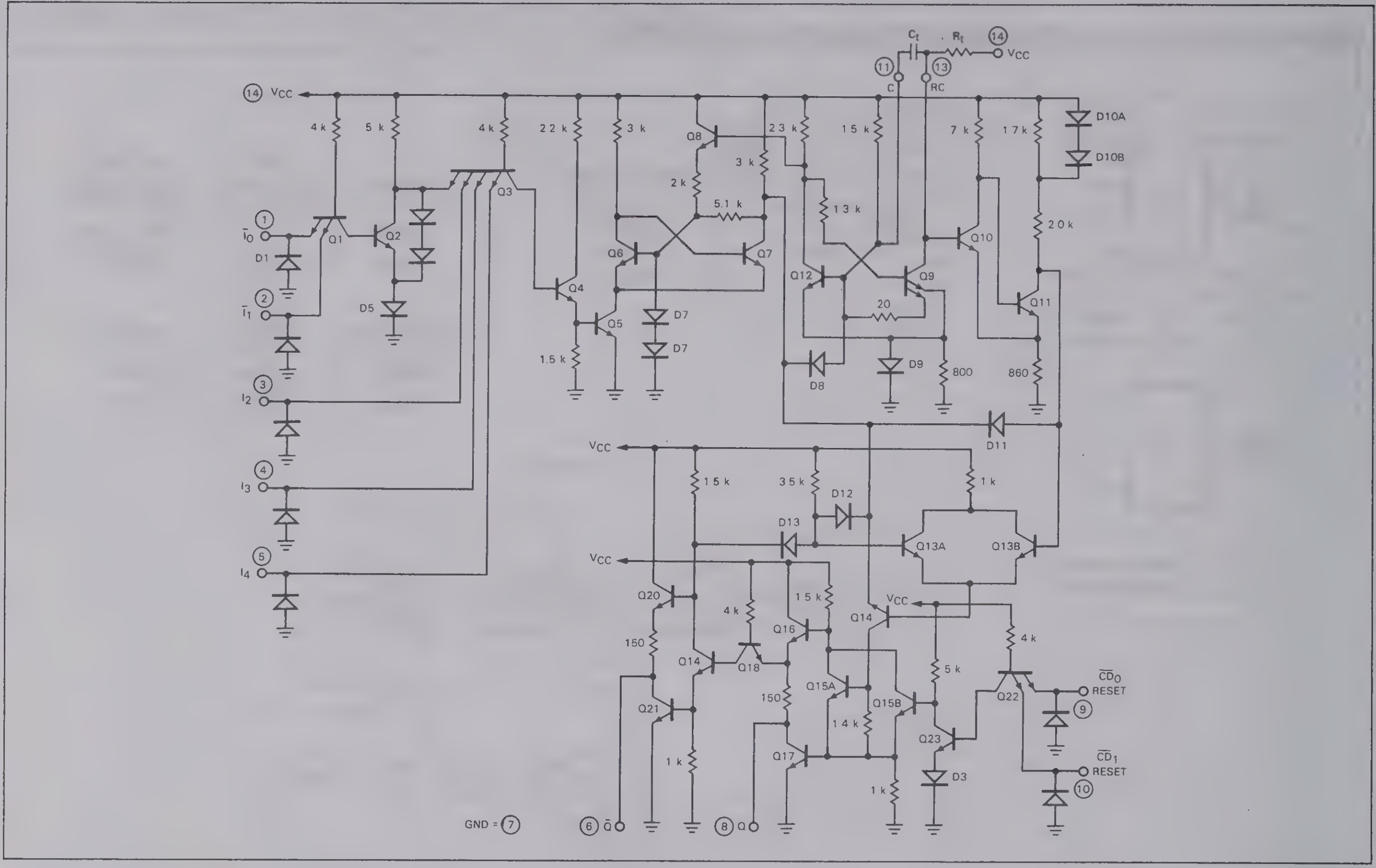
OPERATION	D	S	A	B	C	E	CD	Q	$\overline{Q}$	COMMENT
Quiescent State	H	H	H	H	H	H	L	L	H	
Differentiator Starts	L	H	H	H	H	H	L	L	H	Triggered
	L	H	L	L	H	L	L	L	H	—Start Dis-
	L	H	L	L	L	L	L	H	H	charge Cycle
	L	H	L	L	L	L	L	H	L	
Schmitt Trigger Turns Off	L	L	L	L	L	L	L	H	L	
Differentiator Restores	H	L	L	L	L	L	L	H	L	
										Charge Cycle
Schmitt Trigger Turn On	H	H	L	L	L	L	L	H	L	End of Cycle
	H	H	H	L	L	L	L	H	L	
	H	H	H	L	H	L	L	H	L	
	H	H	H	L	H	H	L	H	L	
	H	H	H	L	H	H	L	L	L	
	H	H	H	L	H	H	L	L	H	
Quiescent State	H	H	H	H	H	H	L	L	H	

The 9602/96L02 circuits are essentially the same as the 9600, except that they are dual, and are similar to the 9601 except for the following.

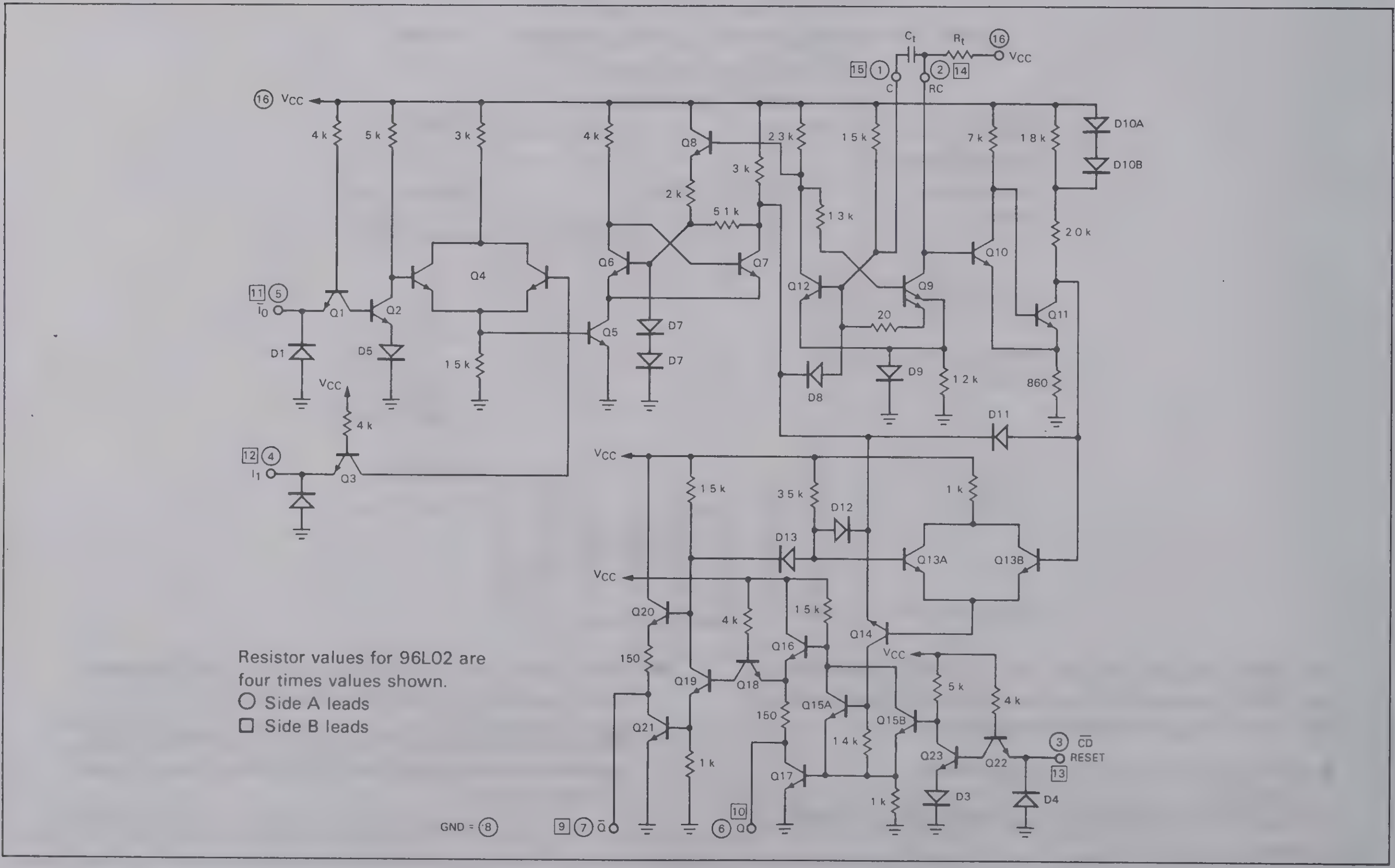
- Different input gating arrangements
- Improved temperature compensation on Schmitt trigger
- Added gate circuitry for reset inputs and feedback from $\overline{Q}$ output to allow reset operation
- 96L02 incorporates very stable temperature compensation to yield less than 1.6% timing change over 0 – 75°C range.

9600/9602/96L02 RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATORS

9600


Ct
Rt
14 VCC
11 C
13 RC
14 VCC
4 k
5 k
4 k
22 k
3 k
23 k
15 k
7 k
17 k
D10A
D10B
Q8
3 k
2 k
5.1 k
13 k
20 k
1 Ī0
D1
Q1
Q2
Q3
Q4
Q5
Q6
Q7
Q9
Q10
Q11
Q12
2 Ī1
D5
D7
D7
1.5 k
20
D8
D9
800
860
3 I2
4 I3
5 I4
D11
VCC
15 k
35 k
1 k
D12
D13
Q13A
Q13B
VCC
Q20
4 k
15 k
Q14
VCC
4 k
150
Q14
Q18
Q16
Q15A
Q15B
5 k
Q22
C̄D0 RESET
9
150
14 k
Q21
Q23
1 k
Q17
D3
C̄D1 RESET
10
1 k
GND = 7
6 Q̄
8 Q

1/2 DUAL 9602/96L02


Ct
Rt
16 VCC
15 1 C
2 14 RC
16 VCC
4 k
5 k
3 k
4 k
Q8
23 k
15 k
7 k
18 k
D10A
D10B
3 k
2 k
5.1 k
13 k
20 k
Q4
Q6
Q7
Q12
Q9
Q10
Q11
11 5 Ī0
Q1
Q2
D1
D5
15 k
Q5
D7
D7
20
D8
D9
12 k
860
VCC
4 k
12 4 I1
Q3
D11
VCC
15 k
35 k
1 k
D12
D13
Q13A
Q13B
VCC
Q20
4 k
15 k
Q14
150
Q16
VCC
Q19
Q18
5 k
4 k
150
Q15A
Q15B
Q21
1 k
14 k
Q23
Q22
3 C̄D RESET 13
Q17
1 k
D3
D4
GND = 8
9 7 Q̄
10 6 Q

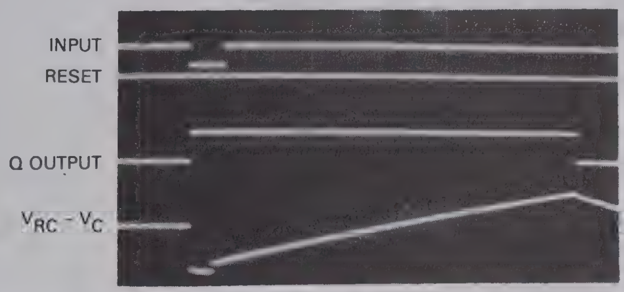
Resistor values for 96L02 are four times values shown.

○ Side A leads

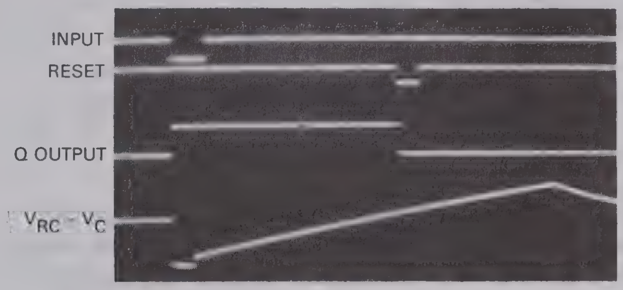
☐ Side B leads

9600/9602/96L02 RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATORS

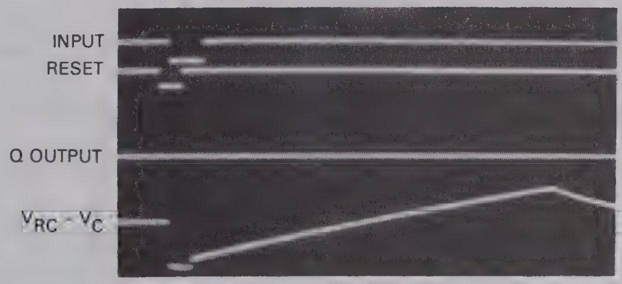
9600/9602/96L02 OPERATION



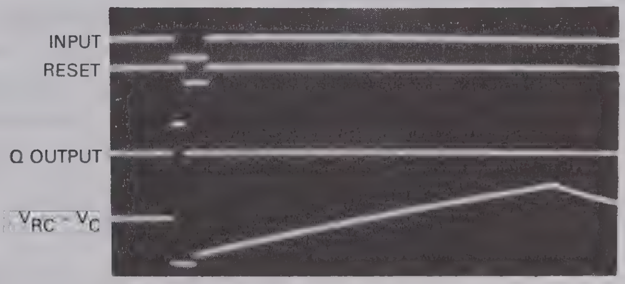
a. RESET NOT ENABLED



b. EARLY TERMINATION OF CYCLE



c. INHIBIT OF DISPLAY OF OUTPUT CYCLE



d. EARLY TERMINATION OF CYCLE

The 9600 and 9602/96L02 monostable multivibrator circuits are similar to the 9601 except for the following:

- Different input gating arrangements, Q1 – Q4
- Modified temperature compensation on Schmitt trigger, D10A, B
- Added gate circuitry for reset inputs, Q22, Q23, Q15B, and feedback from $\overline{Q}$ output to allow reset operation, D12, D13, Q13A.

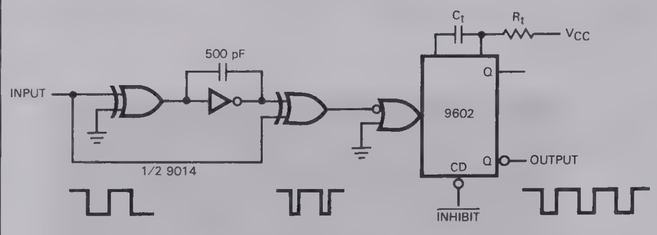
The operation of the 9600 and 9602/96L02 follows the same timing cycle as the 9601. Triggering initiates a discharge cycle, then a charge cycle begins and continues until the Schmitt trigger fires to end the timing cycle (*a* above). The Reset Logic input Q22, Q23 and the added transistor across the output drive transistor Q15B allow the outputs to be held in the Low state (Q = L, $\overline{Q}$ = H) by applying a logic level Low to the Reset input. D13 and the added Q13A transistor allow the outputs to be held Low after the Reset input is restored to a logic High, even if the discharge/charge cycle has not yet completed (Q11 still ON).

Thus the Reset input may be used to terminate the timing cycle before normal time out (*b* and *c* above) or to inhibit a new cycle (*d*). Note that the Reset input does not affect the discharge/charge cycle, but instead inhibits the Q and $\overline{Q}$ outputs from displaying the timing cycle. Retriggering of the 9600, 9602 occurs in the same manner as in the 9601, but the Reset input must not be enabled, i.e., must be at a logic level High, at the time the new trigger pulse occurs. The outputs are set to a High state (Q = H, $\overline{Q}$ = L) only during the time the differentiator flip-flop is on (Q7 ON). The Reset input must not be enabled during this time, otherwise the outputs will remain in the Low state (Q = L, $\overline{Q}$ = H) while the discharge/charge timing cycle continues (*d*). The action of the output latch may be considered as a reset dominant Set/Reset (RS) flip-flop. The Internal Logic diagrams and Internal State Sequence Tables show the internal logic operation of the devices.

9600/9602/96L02 RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATOR APPLICATIONS

The 9600/9602/96L02 monostables may be used in most of the applications shown for the 9601. Exceptions: Schmitt trigger and unstoppable oscillator.

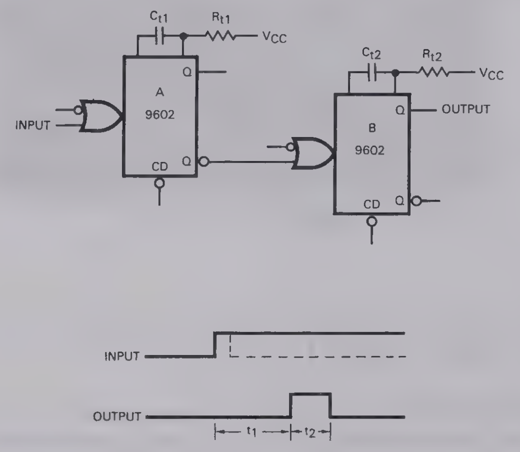
DUAL EDGE TRIGGERING



INPUT
500 pF
1/2 9014
C_t
R_t
V_{CC}
Q
9602
CD
Q
OUTPUT
INHIBIT

This scheme uses 1/2 of a 9014 Quad Exclusive OR/NOR gate to allow the 9602 to trigger on L→H and H→L transitions of the input. The first Exclusive OR gate with the attached inverter (shunted by a capacitor) acts as a delay network. The second Exclusive OR gate compares the input data with the delayed input data signal and produces a Low when the two are different.

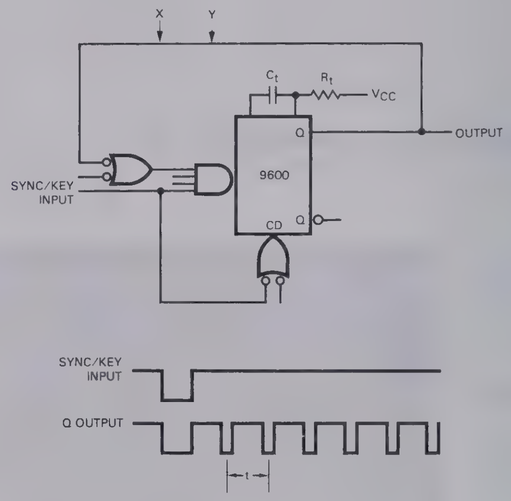
DELAYED PULSE GENERATION



C_{t1}
R_{t1}
V_{CC}
INPUT
A
9602
Q
CD
Q
C_{t2}
R_{t2}
V_{CC}
OUTPUT
B
9602
Q
CD
Q
INPUT
OUTPUT
t_1
t_2

One 9602 connected as shown above provides a means of delaying an input pulse and varying the output pulse width. The first half of this dual one shot determines the time t_1 before the initiation of the output pulse, and the second half determines the output pulse width. While the timing cycle is active, $\overline{Q}$ output of side A is Low. The $\overline{Q}$ output Low-to-High transition at the end of the timing cycle triggers side B. By returning the $\overline{Q}$ output of side B to the input of side A, a free running oscillator with variable duty cycle output is achieved. Due to the edge triggered nature of this oscillator, if an edge is missed, oscillation locks up. If independent adjustment of on and off periods is not required, the 9601 unstoppable oscillator circuit is preferred.

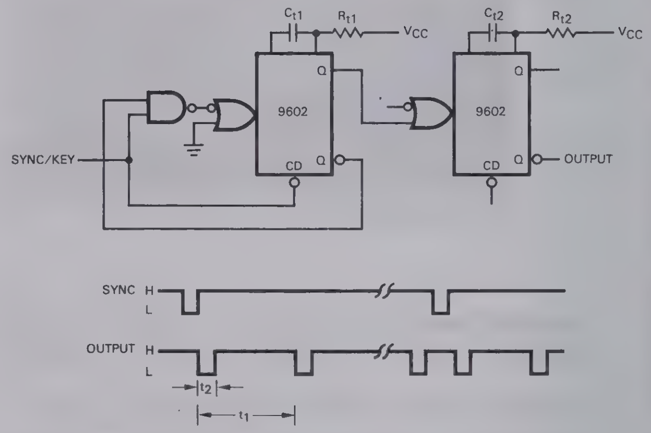
KEYED OSCILLATOR



X
Y
C_t
R_t
V_{CC}
Q
OUTPUT
SYNC/KEY INPUT
9600
CD
Q
SYNC/KEY INPUT
Q OUTPUT
t

The reset feature of the 9600 and 9602/96L02 monostables may be used to construct keyed or synchronizable oscillators. A non-inverting delay network may be inserted between points X and Y to increase the output pulse width.

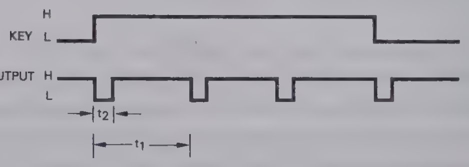
KEYED OR SYNCHRONIZED OSCILLATOR WITH CONSTANT OUTPUT PULSE WIDTH



C_{t1}
R_{t1}
V_{CC}
C_{t2}
R_{t2}
V_{CC}
9602
Q
CD
Q
9602
Q
CD
Q
OUTPUT
SYNC/KEY
SYNC H
L
OUTPUT H
L
t_2
t_1

$$t_1 = 0.31\,R_{t1}\,C_{t1}\left(1 + \frac{1}{R_{t2}}\right)$$

$$t_2 = 0.31\,R_{t2}\,C_{t2}\left(1 + \frac{1}{R_{t1}}\right)$$

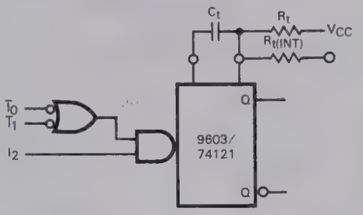


H
KEY L
OUTPUT H
L
t_2
t_1

The 9602 may be used with a 1/4 9002 (1/4 7400) NAND gate to produce a synchronized (keyed) oscillator with constant output pulse width determined by the timing network on the second one shot in the 9602.

9603/74121 MONOSTABLE MULTIVIBRATOR

DESCRIPTION

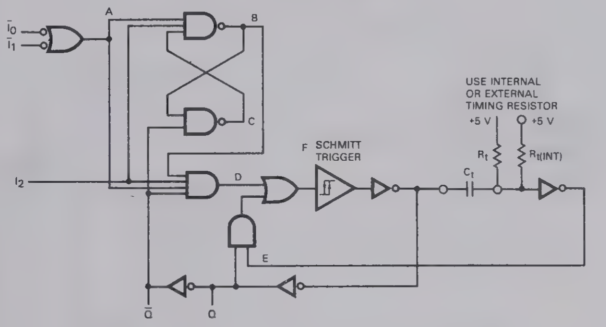

Ct
Rt
VCC
Rt(INT)
$\overline{I_0}$
$\overline{I_1}$
I_2
Q
9603/
74121
Q

$T = (\overline{I_0} + \overline{I_1}) \bullet I_2 \bullet \overline{Q}$ (see note)

$\overline{I_0}$	$\overline{I_1}$	I_2	OPERATION
L	X	L→H	Trigger
X	L	L→H	Trigger
H→L	H	H	Trigger
H	H→L	H	Trigger

NOTE: Triggering occurs only when the $\overline{Q}$ output is H (not in timing cycle) and one of the above truth table conditions is met.

INPUTS	LOADING
High	1 UL
Low	1 UL
OUTPUTS	**DRIVE**
High	10 UL
Low	10 UL


$\overline{I_0}$
$\overline{I_1}$
A
B
C
I_2
D
E
F
SCHMITT
TRIGGER
USE INTERNAL
OR EXTERNAL
TIMING RESISTOR
+5 V
+5 V
Rt
Rt(INT)
Ct
$\overline{Q}$
Q

INTERNAL STATE SEQUENCE
(Note: Delays exaggerated to show sequence of operation)

Refer to Internal Logic Diagram

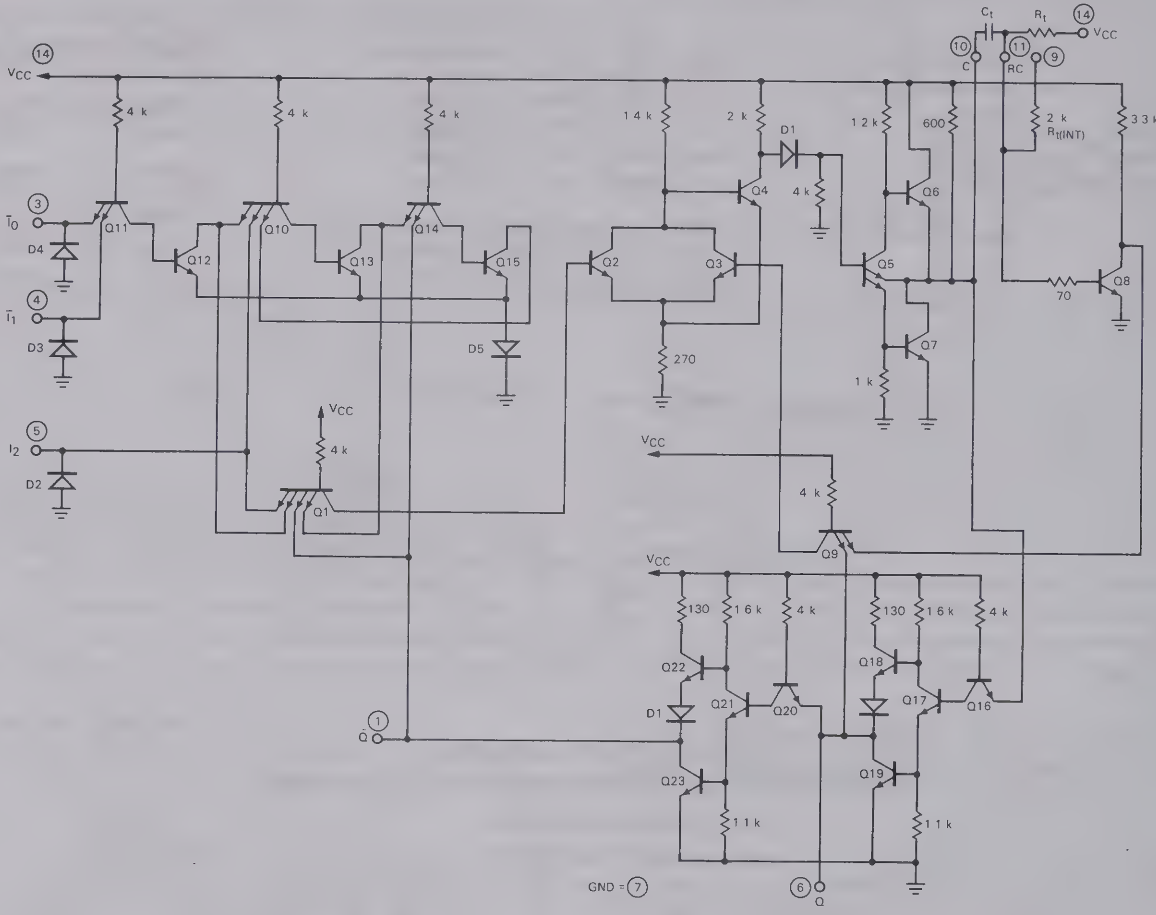
OPERATION	I_2	A	B	C	D	E	F	Q	$\overline{Q}$	COMMENT
	L	L	H	L	L	L	L	L	H	1 Ready States
	H	L	H	L	L	L	L	L	H	2
	L	H	H	L	L	L	L	L	H	3
I_2 goes H	H	H	H	L	L	L	L	L	H	Trigger — (from
	H	H	H	L	H	L	L	L	H	quiescent state 3)
	H	H	H	L	H	L	H	L	H	Start timing
	H	H	H	L	H	H	H	H	H	
	H	H	H	L	H	H	H	H	L	
	H	H	H	H	L	H	H	H	L	— Triggers locked
	H	H	L	H	L	H	H	H	L	out for remainder of cycle
										Capacitor Discharges (Timing Cycle)
E goes L	H	H	L	H	L	L	H	H	L	— Capacitor
	H	H	L	H	L	L	L	H	L	charges
	H	H	L	H	L	L	L	L	L	
	H	H	L	H	L	L	L	L	H	— End Cycle
I_2 goes L	L	H	L	H	L	L	L	L	H	— Remove trigger
	L	H	H	H	L	L	L	L	H	signal
	L	H	H	L	L	L	L	L	H	— Ready to accept new trigger (ready state 3)

The 9603/74121 is a TTL monostable multivibrator with dc triggering from positive or gated negative going inputs and with inhibit facility. Both positive and negative going output pulses are provided with full fan out to 10 normalized unit loads. There are four subcircuits.

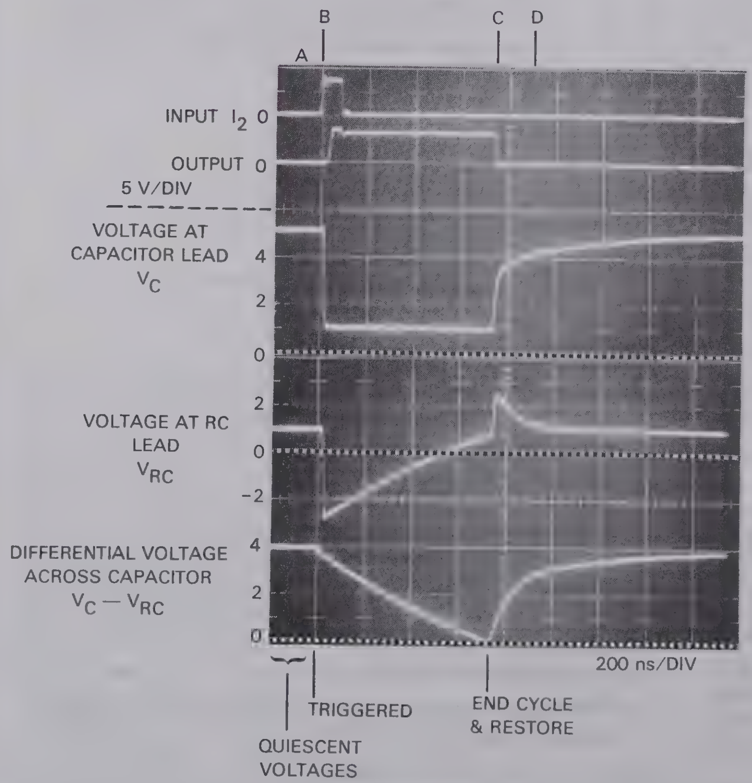
- Input trigger logic
- Input Schmitt trigger
- Timing capacitor charge and discharge circuit and threshold detection
- High and Low output circuits

9603/74121 MONOSTABLE MULTIVIBRATOR

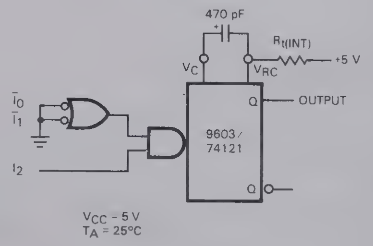
OPERATION



14
VCC
4 k
4 k
4 k
3
$\bar{I}_0$
Q11
Q10
Q14
Q12
Q13
Q15
D4
4
$\bar{I}_1$
D3
D5
5
I_2
D2
VCC
4 k
Q1
1 4 k
2 k
D1
4 k
Q4
Q2
Q3
270
1 2 k
600
Q6
Q5
Q7
1 k
Ct
Rt
14
VCC
10
C
11
RC
9
2 k
Rt(INT)
3 3 k
70
Q8
VCC
4 k
Q9
VCC
130
1 6 k
4 k
130
1 6 k
4 k
Q22
Q21
Q20
Q18
Q17
Q16
D1
1
$\bar{Q}$
Q23
Q19
1 1 k
1 1 k
GND = 7
6
Q



CIRCUIT USED TO GENERATE TYPICAL WAVEFORMS



470 pF
Rt(INT)
+5 V
VC
VRC
$\bar{I}_0$
$\bar{I}_1$
I_2
Q
OUTPUT
9603/74121
$\bar{Q}$
VCC = 5 V
TA = 25°C

9603/74121 MONOSTABLE MULTIVIBRATOR

OPERATION

The 9603/74121 Monostable has four subcircuits.

- Input trigger logic Q1, Q11 — Q15
- Input Schmitt trigger Q2, Q4, which allows jitter-free triggering on input rise time as great a 1 V/s
- Timing capacitor charge and discharge circuit, Q5 – Q7 and threshold detection Q8
- High and Low output circuits, Q16 – Q19, Q20 – Q23

Trigger inputs $\overline{I}_0$ and $\overline{I}_1$ are active Low while I_2 is active High. To trigger the one shot, $\overline{I}_0$ or $\overline{I}_1$ must go Low while I_2 is High or I_2 must go High while either $\overline{I}_0$ or $\overline{I}_1$ is Low. Triggering will not occur if $\overline{Q}$ is Low, i.e., during the timing cycle. A Schmitt trigger circuit Q2, Q4 is provided on the I_2 input so that very slow rise-time signals may be used to trigger the one-shot without causing output jitter.

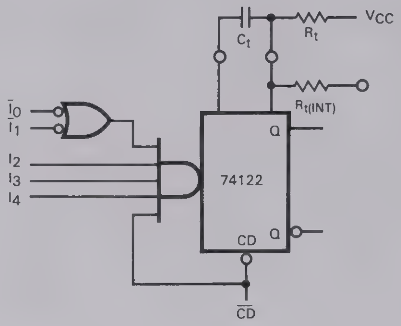
Before an input trigger occurs (i.e., $\overline{I}_0$ or $\overline{I}_1$ is Low, I_2 is Low and $\overline{Q}$ = High), the Schmitt trigger output transistor Q4 is on, since Q2 is held off by the collector of Q1, and Q3 is held off by the Q output through Q9. With Q4 on, Q5 is off and thus Q6 is on and Q7 is off. The 600 Ω resistor pulls the C lead to V_{CC} (voltage's are shown at point A on waveform). Q17 is then held on by base drive through the base collector junction of Q16. Since Q17 is on, the Q output is Low. Q21 is off, thus the $\overline{Q}$ output is High.

When Q7 turns on (point B on waveform), both C and RC leads are pulled down and the timing capacitor is allowed to discharge through the timing resistor and the collector of Q7. Since the RC lead is now pulled down through C_t, Q8 goes off, allowing Q3 to turn on (base drive supplied through the base collector junction of Q9), and thus Q4 is held off during the remainder of the timing cycle. The timing capacitor continues to discharge until the base emitter junction of Q8 becomes forward biased, causing Q8 to turn on (point C on waveform). When Q8 turns on, the base drive to Q3 is removed via Q9, causing Q4 to turn on again. Q4 turns off Q5, and Q6 turns on while Q7 turns off. Turning Q6 on allows the Q output to return to a Low state and the $\overline{Q}$ to go High, while C_t begins charging through the emitter collector of Q6, the 70 Ω resistor and the base emitter junction of Q8. The 600 Ω resistor allows C_t to continue charging when the voltage at the C lead is less then $V_{CC} - V_{BE(Q6)}$ (point D on waveform). Q8 is held on by base current through the timing resistor R_t.

When the $\overline{Q}$ output returns to High at the end of the timing cycle, the input gating logic is ready to accept a new trigger pulse. If a new trigger pulse occurs before the timing capacitor is fully charged to its quiescent 4 V, the output timing cycle will be less than that predicted by the timing formula, $t = R_tC_t\log_e 2$. If a 2 kΩ timing resistor is used, the recommended maximum duty cycle is 66%. Duty cycles of 90% may be achieved by using a 40 kΩ timing resistor. Even higher ratios are obtainable if some pulse width degradation is acceptable.

74122 RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATOR

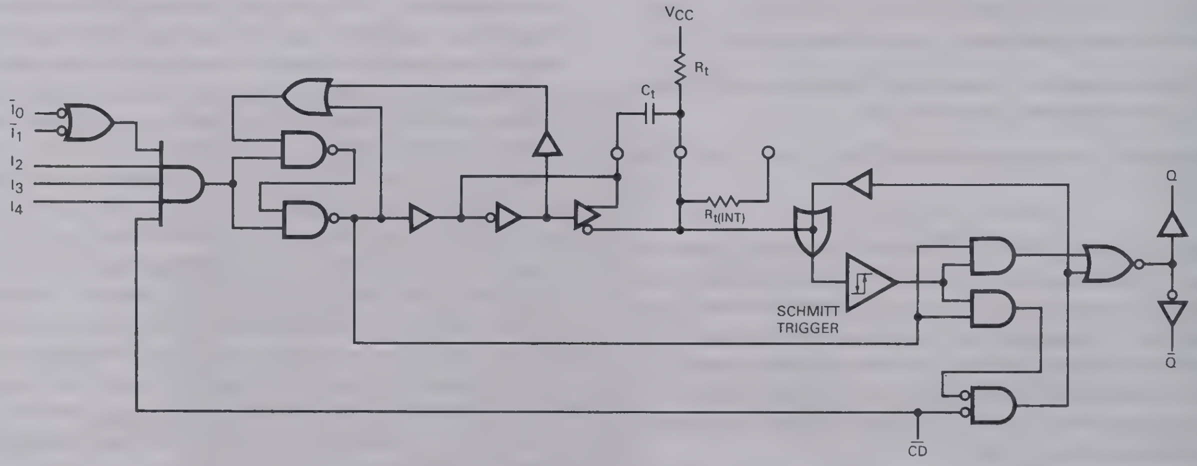
DESCRIPTION AND OPERATION



VCC
Ct
Rt
Rt(INT)
$\overline{I_0}$
$\overline{I_1}$
I_2
I_3
I_4
Q
74122
CD
$\overline{Q}$
$\overline{CD}$

LEADS		LOADING	
$\overline{I_0}, \overline{I_1}, I_2, I_3, I_4$	Data Inputs	1 UL	
$\overline{CD}$	Clear Input	2 UL	
		HIGH	LOW
Q, $\overline{Q}$	Outputs	20 UL	10 UL

$\overline{I_0}$	$\overline{I_1}$	I_2	I_3	I_4	$\overline{CD}$	Response
H→L	H	H	H	H	H	Trigger
H	H→L	H	H	H	H	Trigger
L	X	L→H	H	H	H	Trigger
X	L	L→H	H	H	H	Trigger
L	X	H	L→H	H	H	Trigger
X	L	H	L→H	H	H	Trigger
L	X	H	H	L→H	H	Trigger
X	L	H	H	L→H	H	Trigger
X	X	X	X	X	H→L	Reset
X	X	X	X	X	L	Reset
L	X	H	H	H	L→H	Trigger
X	L	H	H	H	L→H	Trigger



VCC
Rt
Ct
$\overline{I_0}$
$\overline{I_1}$
I_2
I_3
I_4
Rt(INT)
SCHMITT TRIGGER
Q
$\overline{Q}$
$\overline{CD}$

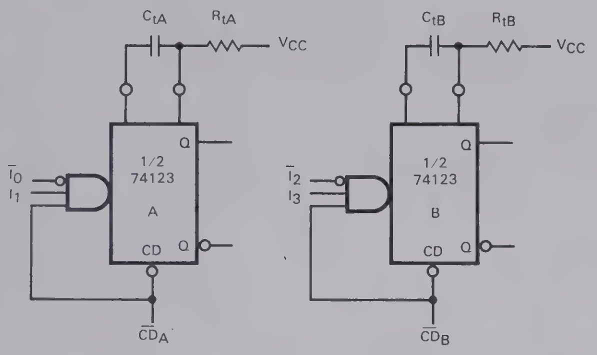
The 74122 is similar to the 9600 but there are important differences in the operation of Reset input $\overline{CD}$.

In the 9600, the trigger logic and the reset function are completely separated. The device triggers only when the value of the trigger logic equation goes from False to True. It resets either with normal time out or when a logic Low is applied. Returning $\overline{CD}$ to High does *not* cause the 9600 to trigger.

In the 74122, the $\overline{CD}$ is internally tied into the trigger network, therefore, the 74122 triggers when the $\overline{CD}$ input is returned to the High state, provided the value of the trigger logic equation is True (see last two lines of the triggering truth table). This feature may be useful in a keyed or synchronized oscillator circuit; but it is undesirable and even unacceptable in the majority of applications. For new designs the 9600 is recommended.

74123 DUAL RETRIGGERABLE RESETTABLE MONOSTABLE MULTIVIBRATOR

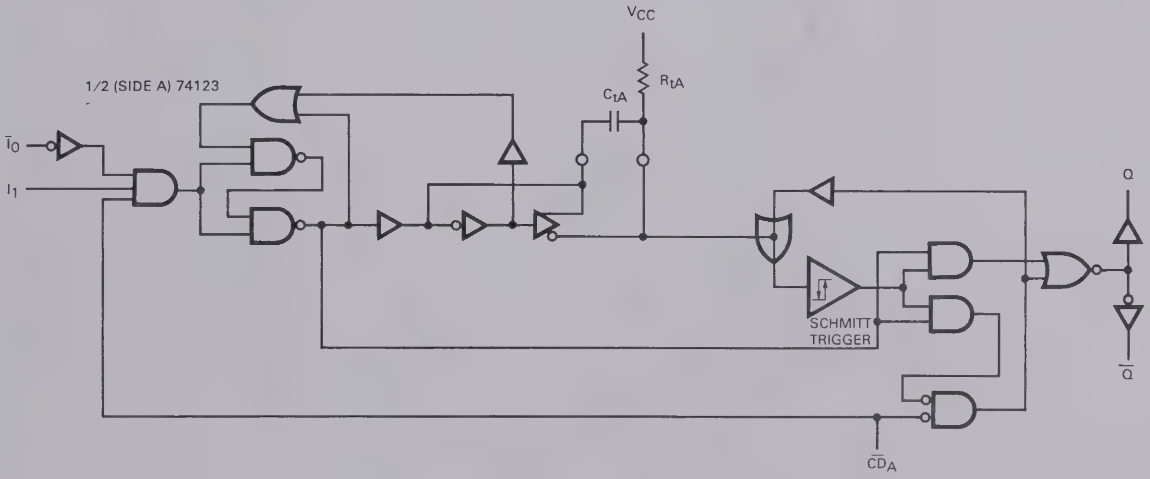
DESCRIPTION AND OPERATION



LEADS		LOADING	
$\overline{I}_0, I_1, \overline{I}_2, I_3$	Data Inputs	1 UL	
CD	Clear Input	2 UL	
		HIGH	LOW
Q, $\overline{Q}$	Outputs	20 UL	10 UL

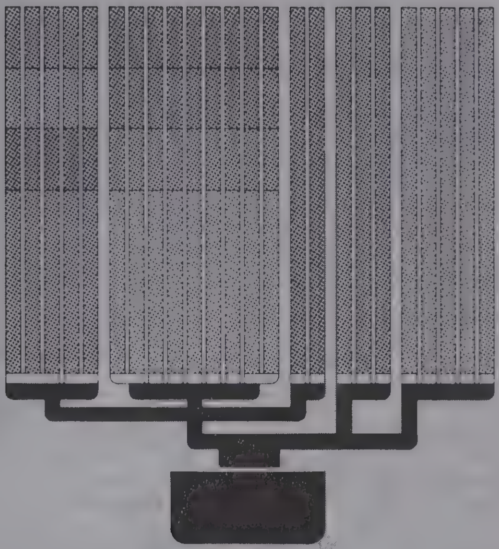
$\overline{I}_0$	I_1	$\overline{CD}_A$	Response
H→L	H	H	Trigger
L	L→H	H	Trigger
X	X	H→L	Reset
X	X	L	Reset
L	H	L→H	Trigger

$\overline{I}_2$	I_3	$\overline{CD}_B$	Response
H→L	H	H	Trigger
L	L→H	H	Trigger
X	X	H→L	Reset
X	X	L	Reset
L	H	L→H	Trigger



The internal circuitry of the 74123 is the same as the 74122 except for reduced number of trigger inputs per one shot, and the absence of the internal resistors $R_{t(int)}$. The 74123 triggers when the $\overline{CD}$ input is returned to the High state provided the value of the trigger logic equation is True. If it is necessary to remove the $\overline{CD}$ signal without triggering under any input logic condition, the 9602 or 96L02 should be used. The 9602 and 96L02 do not have any internal connections between the $\overline{CD}$ and input gating logic, so the triggering and resetting functions are logically and physically independent.

An additional problem arises when the 74123 is used in a non-retriggerable mode. The only safe method of providing non-retriggerable operation under all input conditions is to force the trigger network to be True during the timing period. This is not possible with the AND input structure of the 74123. The only possible connection of inhibiting trigger inputs during the time period results in an astable (free running) operation, if the trigger input level is still active at the end of the timing period. The OR input structure of the 9602/96L02 allows the trigger network to be forced True during time out, which avoids astable operation. The 74123 is not recommended for use where non-retriggerable operation is desired, unless the triggering signal is assured to be False when the 74123 times out. The 9602 or 96L02 is recommended where non-retriggerable operation is desired, due to freedom from possible astable oscillation with the non-retriggerable feedback connection. In new designs, the 9602 or 96L02 is recommended instead of the 74123.



Transmission Line Interface Elements

Line Matching
Unterminated Effects
Parallel Termination
Series (Backmatched) Termination
Long Transmission Line Effects
Data Signal Quality
Line Length vs Data Rate vs Signal Quality
Selecting Line Drivers and Receivers
Simplex Systems
Multiplex Systems
Single Ended Circuits
Balanced Differential Circuits
Half-Duplex Differential Application
EIA RS 232-C Interface
MIL STD 188C Interface
IBM 360/370 I/O Interface

TRANSMISSION LINE INTERFACE SELECTION GUIDE

*SW — Single Wire over Ground Connection
TP — Twisted Pair
TPS — Shielded Twisted Pair
COAX — Coaxial Cable

Standard

Interface	Recommended Driver/Receiver		Comments
EIA RS 232-C	9616	9617	0 to 20,000 bps, maximum cable length implied in standard is 50′.
MIL STD 188C	9616	9627	Use capacitor from 9616 output to ground to provide wave-shaping at applicable modulation rate.
IBM 360 I/O	8T23	8T24	Recommended maximum of 10 ports on bus.

Single-Ended Simplex

Line Length (feet)	Maximum Data Rate (NRZ Data)	Line* Type and Z_0		Recommended Driver/Receiver		Comments
0 – 2′	20M bps	SW TP COAX	>90Ω	TTL Gate	TTL Gate	Unterminated line. Obey loading rules.
2 – 20′	10M bps	TP TPS COAX	>90Ω	9009 or 7440	TTL Gate	Use parallel terminated line with more than one receiver. Use series terminated line with only one receiver.
		COAX	≥50Ω	9S140	TTL Gate	Use parallel terminated line with more than one receiver. Use series terminated line with only one receiver.
20 – 500′	10M bps@ 20′ 0.5M bps@500′	TPS COAX	≥50Ω	8T13	8T14	Use parallel terminated line.
> 500′						Not recommended. Use balanced differential form to gain system noise immunity.

Single-Ended Multiplex

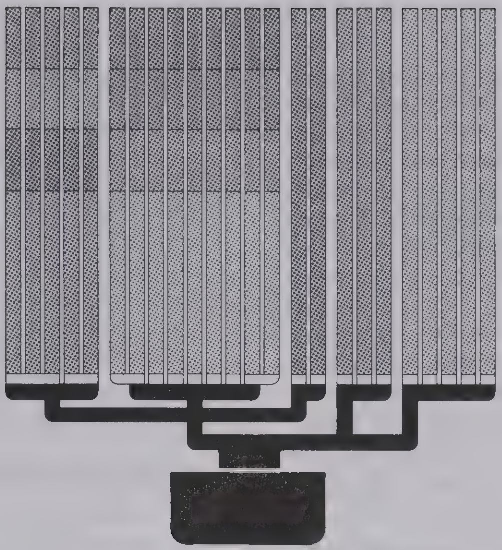
Line Length (feet)	Maximum Data Rate (NRZ Data)	Line* Type and Z_0		Recommended Driver/Receiver		Comments
0 – 2′	10M bps	SW		Open Collector TTL	TTL Gate	Use wired-AND with low value (< 1 kΩ) collector pull up resistor. Obey loading rules.
2 – 20′	10M bps	TP COAX	>75Ω	8T13 or 8T23	8T14 or 8T24	Single +5 V supply. Use parallel termination at both ends of bus.
20 – 500′	10M bps @ 20′ 0.5M bps@500′	COAX	>95Ω	8T23	8T24	Use parallel termination at both ends of bus. Single +5 V supply required.
> 500′						Not recommended. Use balanced differential form to gain system noise immunity.

Balanced-Differential Simplex

Line Length (feet)	Maximum Data Rate (NRZ Data)	Line* Type and Z_0		Recommended Driver/Receiver		Comments
0 – 50′	10M bps	TP TPS	>80Ω	9614	9615	Use parallel termination. Single +5 V supply required.
				75109/110	75107/108	Split parallel termination. +5 and −5 V supplies required.
50 – 4,000′	Use signal quality graph.	TPS	>80Ω	9614	9615	Use parallel termination. Single +5 V supply required.
				75110	75107/108	Use split parallel termination. +5 V and −5 V supplies required.
> 4,000′						Cable loss exceeds 6 dBV. Perhaps non-baseband techniques should be used, (i.e., MODEMS).

Balanced-Differential Multiplex

Line Length (feet)	Maximum Data Rate (NRZ Data)	Line* Type and Z_0		Recommended Driver/Receiver		Comments
0 – 50′	15M bps	TP TPS	>90Ω	75110	75107/108	Use split parallel termination at each end of line. Requires +5 V and −5 V supplies.
		TP TPS	>90Ω	9614	9615	Connect as shown in half duplex differential circuit. Requires single +5 V supply.
50 – 4,000′	Use signal quality graph.	TPS	>90Ω	75110	75107/108	Use split parallel termination at each end of line. Requires +5 and −5 V supplies.
> 4,000′						Cable loss exceeds 6 db V. Perhaps non-baseband techniques should be used, (i.e., MODEMS).



TRANSMISSION LINE INTERFACE ELEMENTS

INTRODUCTION

The interface circuits discussed here are devices which convert TTL signals into signals adapted for transmission lines (line drivers), and devices to convert transmission line signals back into TTL signals (line receivers). TTL logic (SSI and MSI) cannot normally afford the extra power dissipation required to drive low impedance (50 Ω) transmission lines. Regular TTL parts have only limited drive capability, and line driver circuits are needed to provide the transmission line interfacing. The drive capability of the various TTL families is discussed in Small Scale Integration.

TTL can drive ordinary interconnections as long as those interconnect time delays are much less than the TTL Rise or Fall Time. This will reduce the current required to produce a given voltage on the line, and generally cause the transmission line to act more like a simple interconnection. This tradeoff of line delay vs. driver rise/fall time will allow TTL parts to use lower power output stages. This tradeoff will restrict normal TTL to a maximum interconnect distance of approximately 2 feet, which is adequate for device-to-device connections on a PC card or wire-wrap board. Interconnections longer than those recommended will result in more pronounced transmission line effects, and generally will degrade system noise immunity and performance. If long lines, large amounts of external noise, and a bus organized circuit are all required or present simultaneously, then the lack of drive and relatively small noise margins of regular TTL is severely felt. For most purposes, any interconnection longer than two feet should be treated as a transmission line, and line drivers/line receivers should be used.

Fairchild offers a number of line drivers/receivers which vary considerably in configuration, function, complexity and cost. The applications selection guide includes the major performance/physical parameters of all currently available interface devices.

USING THE SELECTION GUIDE

The selection guide at the beginning of the chapter requires three basic decisions — standardized or special interface, single-ended or balanced differential form, simplex or multiplex mode. If a standardized interface is required (EIA RS 232-C Mil Std 188C - low level, or IBM 360 I/O), use the standardized section of the guide.

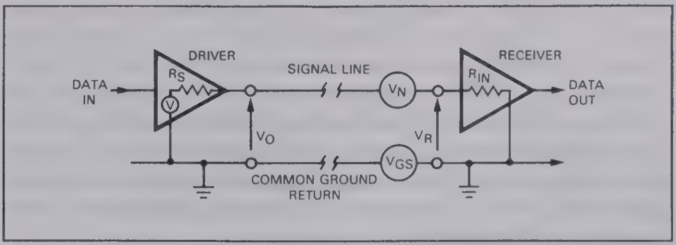
The first decision necessary for a non-standardized interface involves form of operation. Generally, in high noise environments or when long lines are required (>500 ft), the balanced differential form is preferred.

After selecting the form, the mode of operation must be selected. The simplex mode is the easiest to implement and offers fewer potential design difficulties. The multiplex mode allows overall reduction in cable costs when a large number of bits is to be exchanged over transmission lines.

FORMS AND MODES OF OPERATION OF DATA COMMUNICATION CIRCUITS

There are two basic forms of data communication circuits, single-ended and balanced differential. The single-ended circuit uses a signal line and a common ground return.

SINGLE-ENDED CIRCUIT

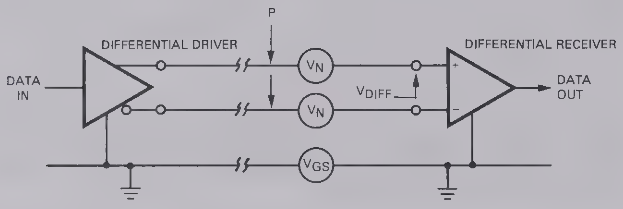


TRANSMISSION LINE INTERFACE ELEMENTS

The advantage of the single-ended system is simplicity with only one signal wire required per circuit. The disadvantage is susceptibility to induced noise V_N and ground shift noise V_{GS}. Induced noise is caused by magnetic and capacitive coupling from adjacent signal lines or other noise generators such as brush-type motors or SCR lamp dimmers. Ground shift noise is the result of the voltage potential developed across a ground circuit with a finite resistance and inductance due to flow of current through the ground. The net effect of V_N and V_{GS} alters the voltage at the receiver input. V_N and V_{GS} are added to any signal produced by the driver, and a receiver cannot discriminate between a legitimate signal and a signal which is the sum of the noise in the circuit and the actual signal produced by the driver. The noise immunity of single-ended circuits, however, can be improved with three methods.

- Use shielded cable and reduce ground impedance which decreases noise but adds cost.
- Increase the induced driver output voltage levels which swamps out noise but increases power consumption.
- Add hysteresis in the receiver which increases the dc noise margin but may introduce time distortion.

BALANCED DIFFERENTIAL CIRCUIT


P
DIFFERENTIAL DRIVER
DIFFERENTIAL RECEIVER
DATA IN
DATA OUT
VN
VN
VDIFF
VGS
+
–

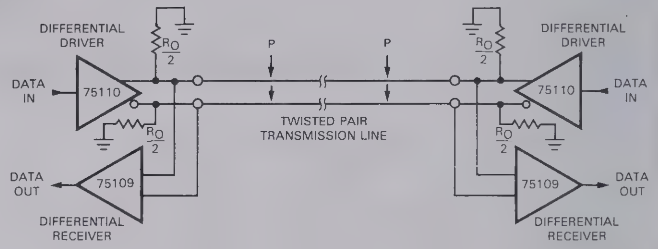
As illustrated, the balanced differential circuit uses a twisted pair of wires as a transmission line, a differential driver, and a differential receiver. The twisted pair transmission line cancels magnetically induced currents in the line because of adjacent twists of the line.

Electrostatically coupled noise equally affects both lines of the twisted pair. Thus, it is transformed into a common mode signal at the receiver. The ground shift noise also appears to the receiver as a common mode voltage. Because the signal voltage V_{DIFF} and the noise voltage $V_N + V_{GS}$ appear to the receiver as differential and common mode voltages, respectively, they can be separated by a receiver with high common mode rejection. In this manner, information can be transferred through environments which would otherwise cause errors in a single-ended system.

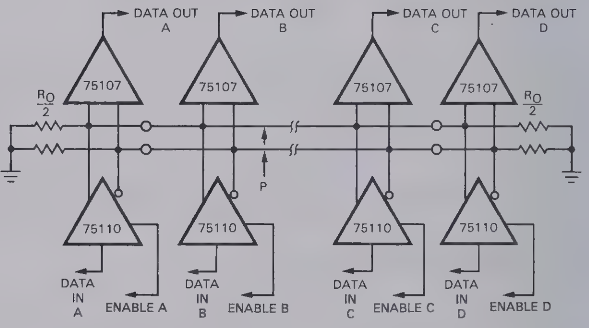
SIMPLEX VERSUS MULTIPLEX

There are two basic modes of operation for data communication circuits: Simplex and multiplex. A simplex circuit allows one way, nonreversible data flow. A multiplex circuit allows bidirectional (half-duplex) or multidirectional (data bus or party line), non-simultaneous data flow. A general multiplex circuit has two or more pairs of drivers and receivers on the same line, but only one driver may be transmitting at any one time. A special multiplex mode is a distribution system that has one driver and two or more receivers placed at various locations on the line. Some basic multiplex modes of operation for data communication circuits are illustrated.

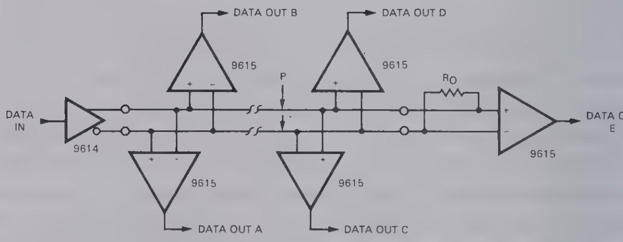
MULTIPLEX MODES OF OPERATION


DIFFERENTIAL DRIVER
RO/2
P
P
RO/2
DIFFERENTIAL DRIVER
DATA IN
75110
75110
DATA IN
TWISTED PAIR TRANSMISSION LINE
RO/2
RO/2
DATA OUT
75109
75109
DATA OUT
DIFFERENTIAL RECEIVER
DIFFERENTIAL RECEIVER

HALF DUPLEX MODE
BALANCED DIFFERENTIAL


DATA OUT A
DATA OUT B
DATA OUT C
DATA OUT D
75107
75107
75107
75107
RO/2
RO/2
P
75110
75110
75110
75110
DATA IN A
ENABLE A
DATA IN B
ENABLE B
DATA IN C
ENABLE C
DATA IN D
ENABLE D

DATA BUS, BALANCED DIFFERENTIAL
(CURRENT SOURCE DRIVERS)


DATA OUT B
DATA OUT D
9615
9615
P
RO
DATA IN
9614
DATA OUT E
9615
9615
9615
DATA OUT A
DATA OUT C

DISTRIBUTION BUS
BALANCED DIFFERENTIAL

Operating mode selection depends mainly on the system requirements. As a general rule, simplex systems are easier to implement because timing problems are minimal. Multiplex systems conserve overall system wire costs but are more difficult to design.

MULTIPLEX OPERATION CONSIDERATIONS

Some of the multiplex mode (data bus) operational methods and problems that must be considered are as follows:

- The protocol or "handshaking" required for a particular port on the bus to send data must be designed. The protocol sequence must usually involve the following operations:

  1) The port must signal a desire to use the bus (interrupt).
  2) Bus controller must acknowledge interrupt and send "go ahead" command.
  3) Port assumes control of bus and sends data, perhaps preceded by the code to indicate the recipient(s) of the following data.
  4) Receiving port(s) must acknowledge receipt of data.
  5) Transmitting port receives the acknowledgement, and releases control of the bus so that other ports may pass their data.

  The overall bus operation is either polled or asynchronous. In polled operation, a central bus controller addresses each port in turn to ask if any data is waiting to be sent. If the addressed port has no traffic, it signals "no data" and the controller inquires at the next port. If the port has some data, then the controller gives a "go ahead" to the port; data is sent, and the controller then inquires at the next port.

  In asynchronous operation, any port having data essentially "holds up its hand" and waits for a "go ahead" signal to ripple down the series enabling logic. The priority for a particular port is determined by the ports proximity to the master control port which sends a "go ahead" down the enable logic chain at regular time intervals. This scheme is well suited to bus-organized minicomputers. The Digital Equipment Corporation's Unibus® and Omnibus® architectures are excellent examples.

- The effect of powered-down drivers and receivers to normal bus operation must also be considered. Integrated circuit drivers and receivers contain parasitic diodes that are normally reverse-biased when the power supply is on. Unless special design techniques are used, these diodes can become forward-biased when the unit is powered-down, thus causing the bus to malfunction.

- The protocol timing must include sufficient time delays to allow for the different port-to-port signal propagation delays.

- Both physical ends of the transmission line comprising a channel for the bus must be terminated to prevent spurious signal levels due to reflections. (See following section on line matching.)

- Stubs or taps from the main transmission line should be kept to a minimum length. A "daisy chain" wiring method is preferable to a tap off method. If stubs must be used, then to cause the least perturbations on the line, the stub length should be controlled such that the propagation delay of the stub is less than 1/8 of the signal rise or fall time at the stub to line connection point.

- If a 3-state driver system is used (Logic Zero, Logic One and "off", or driver in high impedance state), some means must be provided to detect the difference between a driver sending data and the all drivers off condition. In the 2-state bus system, this problem does not occur because a logic Zero (usually a High) indicates either a logic Zero or that no port is currently sending. A logic One (usually a Low) on a 2-state system then indicates a port is transmitting a logic One and the receiver should interpret it as such.

- The data format must also be considered:

  Parallel operation is fast but expensive, since it requires one transmission line and the associated interface for every bit of the word (or byte) transmitted in parallel.

  Serial operation is slower, but requires only one transmission line and interface per port. This saving may, however, be partially offset by the need for a parallel-to-serial converter at the transmitting site, and a serial-to-parallel converter at the receiving site.

  The parallel structure is commonly used for rapid exchange of data over short distances; e.g., within a computer or between computer and peripherals. The serial structure is used for communications over long distances, as between a terminal and its controller.

- A final consideration concerns polled operation of a multiplex system. The amount of time necessary to address and receive acknowledgement from a port must be weighed against the volume of data the ports normally send, and the total number of ports on the bus. If there is a large number of ports on the bus, most of the time might be used by the polling operation with very little time devoted to actual exchange of information. A large number of ports combined with a high relative volume of traffic expected per port can lead to data backing up at each port waiting to be sent, and an overall reduction in information throughput. In a "real time" system where fast response is essential, serious consideration should be given to splitting up a single large bus into several "satellite" busses, each with its own polling controller and protocol with respect to the central bus. Queuing theory can be used to estimate the throughput on a bus structure when many variables including the number of ports, the mean transaction length, and the number of transactions per port per unit time are known.

LINE MATCHING

The purpose of line matching is to reduce or eliminate errors caused by transmission line reflections in data communication circuits. When system operation is not affected, these reflections can be ignored — similar to ignoring contact bounce in a mechanical switch. When data bit duration is long relative to line propagation delay, reflection effects die away in a relatively short time. However, in most transmission line circuits, relections *do* affect the system and there are three basic methods of minimizing reflection affects.

Method I; Control Ratio of Signal Rise Time to Line Delay

Most logic (TTL, ECL, etc.) has very low output impedance and high input impedance relative to the characteristic impedance of normal interconnections (50 – 200Ω). Transmission line theory predicts that this combination, will exhibit large signal overshoot and undershoot at the receiver end of the line, assuming that the signal rise times produced by the driver are short relative to the time delay of the line.

®Digital Equipment Corporation

If, however, the rise time of the signal is greater than twice the one way time delay of the line, a different result is obtained: the ringing amplitude at the receiver is less than that predicted by classical theory. When the driver output signal rise time is four times as great as the time delay of the line, the ringing is considerably reduced and the line input and output waveforms are almost identical, with only a few small perturbations. This results from the superposition of ramps or exponential type waveforms instead of the unit step variety used in classical transmission line theory. To make an unterminated line driven by a voltage source type driver behave in a more civilized manner, one of two things can be done.

- Reduce the maximum allowable length of the interconnection so its time delay is less than one quarter the rise or fall time of the driver ($t_D < t_r/4$).
- Control the slew rate of the driver so the rise time at the line input is greater than four times the time delay of the line (again $t_r > 4t_D$).

Maximum line length limiting is normally used to formulate logic wiring rules, while explicit driver slew rate control is sometimes applied by such standard interfaces as MIL STD 188C or EIA RS 232-C. In any event, t_r-to-t_D ratio limiting with unterminated lines has a serious drawback in that delay times through the data communication circuit are always greater than those obtained using a terminated line with a fast rise and fall time signal. This is due to the delay of the signal to rise above the threshold level of the receiver.

Method II; Parallel Termination

According to transmission line theory, a line terminated by a resistor equal in value to the characteristic resistance of the line does not exhibit reflections. Waves arriving at the termination point pass into the termination resistor without generating a reverse traveling wave, since Kirchoff's laws are satisfied at the line–termination boundary.

With the parallel termination method, multiple receivers can be used on the circuit forming a distribution bus (one driver, many receivers). Each receiver placed along the line sees the full voltage swing produced by the driver onto the line, after a delay time for the signal to propagate from the driver site to the receiver site. As soon as the wave reaches the opposite line end, it is absorbed by the termination, and the line assumes steady-state conditions. The primary disadvantage of the parallel termination method is the power dissipated in the termination resistor, since it does provide a dc load on the driver device. When voltage source type drivers are used, a capacitor can be inserted in the termination network to block the dc, but still provide a termination to the wavefront arriving at the line end. This ac coupled termination works well when the data is a symmetrical signal such as a clock. If random data is transmitted, long strings of like bits allow the line to charge up to the voltage compliance limit of the driver. A subsequent data bit of the opposite polarity will take longer than normal to cross the receiver threshold since it is starting from a greater potential than normal. This will result in time jitter of the recovered data, dependent on the previous data pattern. For this reason, the ac termination approach is not generally recommended unless the resultant time jitter in the data can be tolerated.

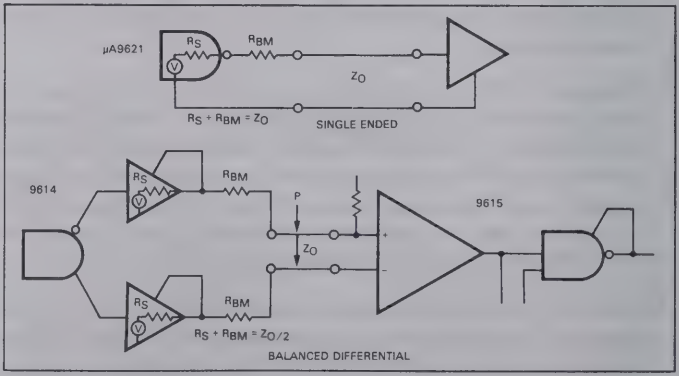
The choice and placement of the parallel termination network depends on the type of driver (voltage or current source) and whether the system is simplex or multiplex mode. For simplex mode with voltage source drivers (9614, 9621 or 8T13, 8T24), a single resistor is placed across the end of the line most distant from the driver. This applies to both single ended and balanced differential lines. In order to reduce the steady state output current required for single-ended systems, a Thevenin equivalent termination can be used; for example, termination of ECL driver lines to –2.0 V through the termination resistor. This is possible with a single resistor connected to a –2.0 V power supply or a 2-resistor voltage divider between –5.2 and ground which produces the Thevenin equivalent. However, more power is consumed by the latter method. Current source drivers such as the 75109/110 series require that each output have a dc path to the common signal return, or ground, so that a split termination is used. If simplex mode differential form is used, the parallel termination network must be two resistors, each equal to half the characteristic impedance of the line connected from each side of the line to the signal common return.

For multiplex operation, *both* ends of the line must have parallel termination networks. This prevents false signal levels from occuring when a driver along the line is enabled. The false signal levels would be caused by a reflection from the unterminated end. The use of single or split resistor termination is again based on the driver types used in the system. Voltage source drivers (8T13 or 8T24) use a single resistor termination, current source drivers (75109,110) require split terminations for balanced differential operation.

Method III; Series Termination or Back Matching

The matched source or backmatched driver method is useful for either single-ended or balanced differential simplex systems where only one driver and one receiver is required. A matched source is constructed by inserting a resistor R_{BM} in series with the voltage source driver output of a value such that the sum of the driver output resistance plus R_{BM} is equal to the characteristic impedance of the line. For differential backmatching with a differential voltage source driver, two resistors are used, one in series with each output. The value of R_{BM} is such that the driver output impedance, plus R_{BM}, equals half the characteristic resistance of the line.

SIMPLEX BACKMATCHED DRIVER SYSTEMS



TRANSMISSION LINE INTERFACE ELEMENT APPLICATIONS

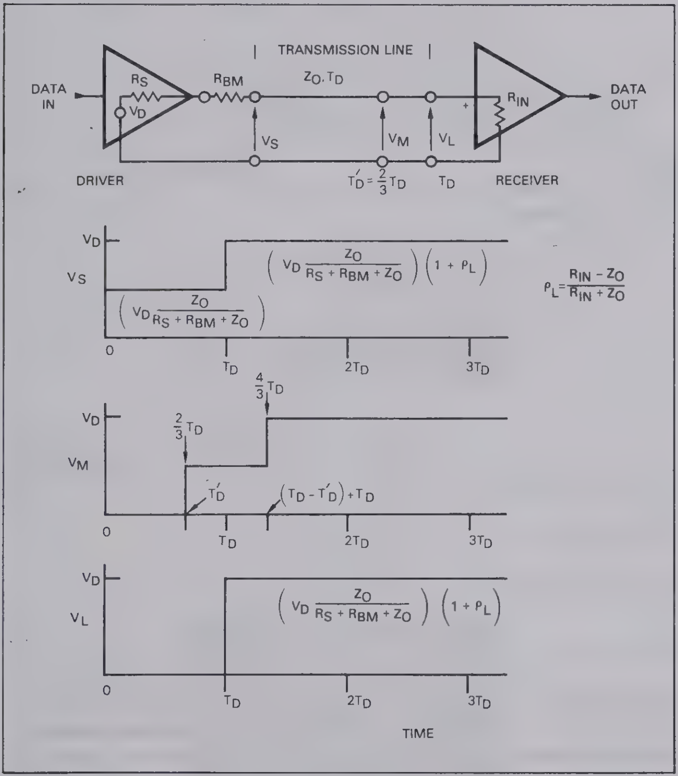
The backmatched driver method is based on two points of transmission line theory.

- An unterminated line approximately doubles the signal level at the receiver site.
- When driver source impedance equals the line impedance, the line appears terminated with respect to signals returning to the source.

Initially, a voltage wave of approximately one half the voltage compliance limit of the driver $V_{DRIVER}\dfrac{Z_O}{Z_O + R_S + R_{BM}}$ is sent toward the receiver. When this wave arrives at the receiver, it sees an impedance discontinuity and a reflected wave is generated at the receiver/line boundary. This reverse wave propagates back toward the source. Its magnitude is ρ_{LOAD} times the arriving wave. ρ_{LOAD} is the voltage reflection coefficient of the load. It is defined as

$$\rho_{LOAD} = \frac{R_{LOAD} - Z_O}{R_{LOAD} + Z_O}$$

LINE VOLTAGES WITH A BACKMATCHED DRIVER



Since the principle of superpostion applies, the voltage at any point on the line is the sum of all waves that have passed that point plus any initial voltage conditions. As the reflected wave travels toward the source, its voltage is added to the residual voltage of the initial wave. When the reflected signal returns to the source, however, no additional reflection is generated.

It is apparent that any additional receivers placed along a transmission line do not receive a full signal swing until the initial voltage wave from the load end returns to their bridging points. As a result these additional receivers have severely reduced noise immunity and may falsely indicate the received data for as long as $2T_D$ (for a receiver located at the driver/line boundary). Thus, the backmatched driver method is limited to:

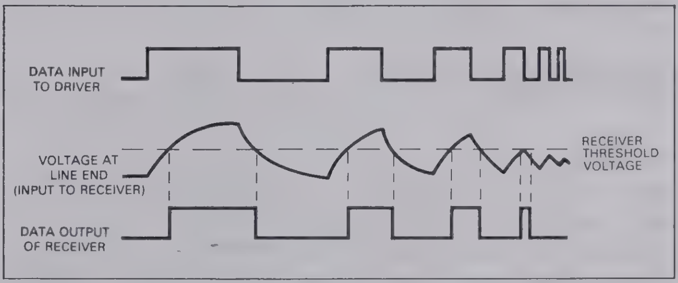
- Simplex mode operation only, and
- Only one receiver located at the end of the line opposite the driver end of the line.

Current source drivers (75109/110) with a split parallel termination at the driver site produce a backmatched driver. Likewise, the open emitter voltage source drivers (8T13/23) produce a backmatched source when the parallel termination resistor is located at the driver site.

LONG TRANSMISSION LINE EFFECTS

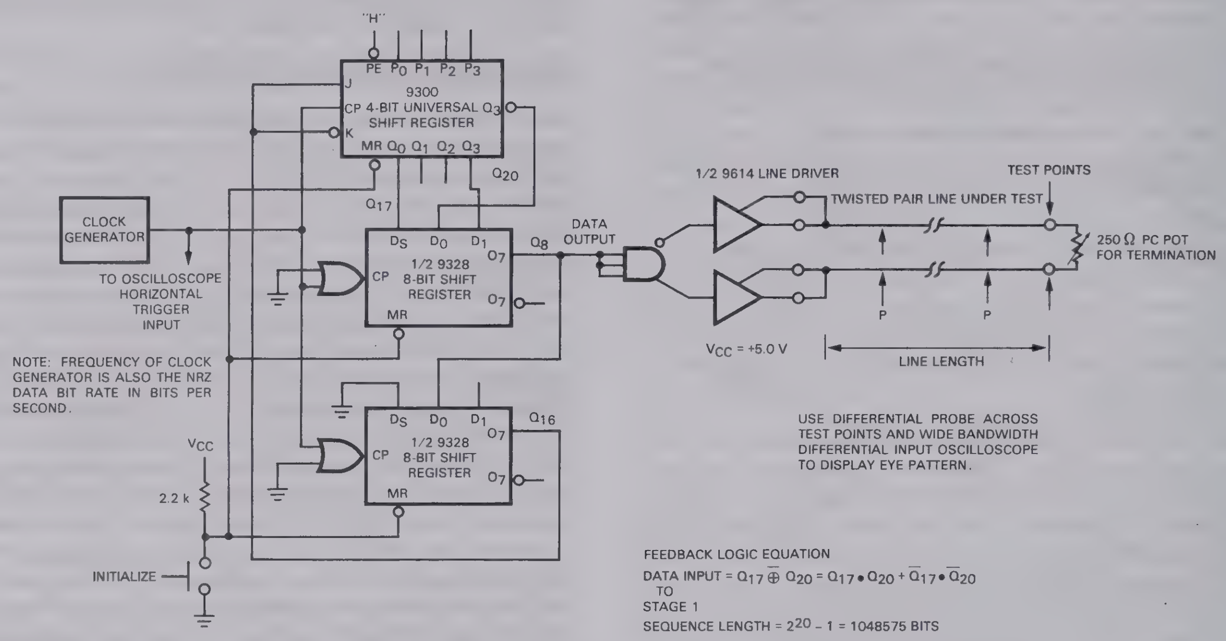
The last characteristic considered here on data communications circuits is the effect that a long transmission line has on the signals passing through it. In many ways, a long transmission line resembles a low pass filter with loss. A fast signal transition applied to the input of the line becomes more rounded and exponential as the signal propagates down the line. Additionally, the resistance in the wires comprising the line causes an overall reduction in signal amplitude. These two effects combine to limit the data rate that can be carried by a given type and length of line. Of these two effects, the change in signal waveshape is the most limiting to the maximum data rate. If a new data bit is sent and arrives at the line end before the line has completed it's response to the previous data bit, then the signal representing the new data bit can cross the line receiver's threshold earlier than normal. This causes the recovered data from the line receiver to exhibit pattern dependent transition displacement, or time jitter. In the limit, as a shorter and shorter pulse is sent, the signal at the line end may not even have time to cross the receivers threshold, and the data bit might be missed completely. This phenomenon is called intersymbol interference, and is due to the previous data bit (or bits) causing a time shifting of the threshold crossing of the present data bit. Intersymbol interference starts to occur any time the minimum time duration of a data bit is less than the signal rise (or fall) time at the end of the line. The figure shows this effect.

LINE DISTORTION EFFECT ON DATA BIT WIDTH RESOLUTION


DATA INPUT TO DRIVER
VOLTAGE AT LINE END (INPUT TO RECEIVER)
RECEIVER THRESHOLD VOLTAGE
DATA OUTPUT OF RECEIVER

A clock signal (equivalent to alternating Ones and Zeros) will not show intersymbol interference, since the previous data pattern is highly symmetrical and also predictable. A random data pattern, however, is prone to intersymbol interference; the minimum bit duration occurs from a single bit preceded and followed by a long string of bits of the opposite polarity. Thus, to measure the amount of intersymbol interference present, and estimate the resultant bit transition time jitter, a long random or pseudo-random data sequence must be used.

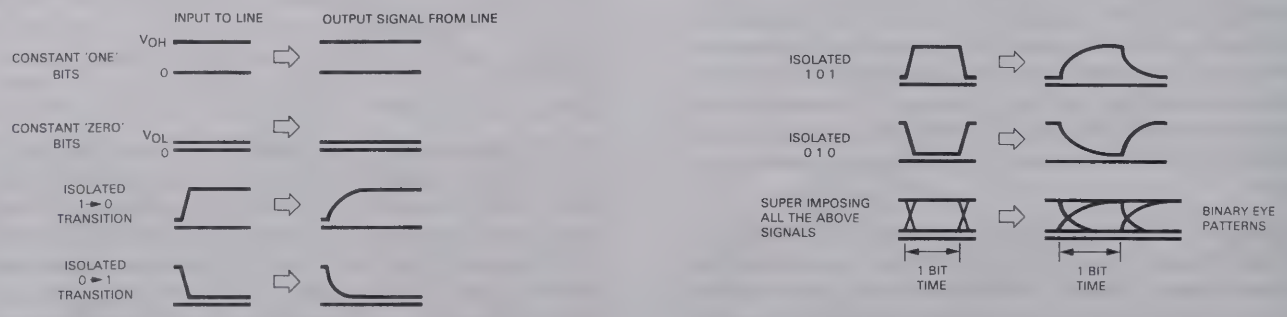
TEST SETUP TO DISPLAY EYE PATTERN FOR MEASURING DATA SIGNAL QUALITY


"H"
PE P0 P1 P2 P3
J
9300
CP 4-BIT UNIVERSAL Q3
SHIFT REGISTER
K
MR Q0 Q1 Q2 Q3
Q20
Q17
CLOCK GENERATOR
TO OSCILLOSCOPE HORIZONTAL TRIGGER INPUT
NOTE: FREQUENCY OF CLOCK GENERATOR IS ALSO THE NRZ DATA BIT RATE IN BITS PER SECOND.
DS D0 D1 O7
1/2 9328
CP 8-BIT SHIFT REGISTER
O7
MR
Q8
DATA OUTPUT
Q16
VCC
2.2 k
INITIALIZE
1/2 9614 LINE DRIVER
TWISTED PAIR LINE UNDER TEST
TEST POINTS
250 Ω PC POT FOR TERMINATION
P
P
VCC = +5.0 V
LINE LENGTH
USE DIFFERENTIAL PROBE ACROSS TEST POINTS AND WIDE BANDWIDTH DIFFERENTIAL INPUT OSCILLOSCOPE TO DISPLAY EYE PATTERN.
FEEDBACK LOGIC EQUATION
DATA INPUT TO STAGE 1 = $Q_{17} \overline{\oplus} Q_{20} = Q_{17} \bullet Q_{20} + \overline{Q}_{17} \bullet \overline{Q}_{20}$
SEQUENCE LENGTH = $2^{20} - 1$ = 1048575 BITS

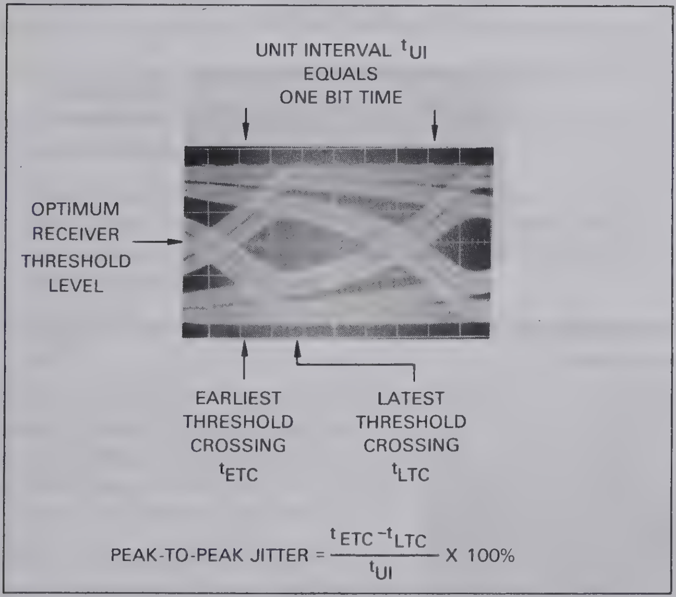
PSEUDO-RANDOM SEQUENCE GENERATOR

A bench setup to display the intersymbol interference is shown in the figure. Since the oscilloscope is triggered each time a new data bit can appear from the Pseudo-Random Sequence Generator (PRSG), the display will show the superposition of long strings of Ones and Zeros, Zero-to-One and One-to-Zero transitions preceded and followed by various data patterns. The 2-device PRSG shown will generate a sequence that will repeat after 2^{20-1} bits; this sequence is sufficiently long to allow display of possible intersymbol interference caused by previous data patterns to a depth of 20 bits. The figure shows how the superposition over one bit interval of the waveforms at the end of the line due to various data patterns will produce a stable display on the oscilloscope.

FORMATION OF A BINARY EYE PATTERN


INPUT TO LINE
OUTPUT SIGNAL FROM LINE
CONSTANT 'ONE' BITS
VOH
0
CONSTANT 'ZERO' BITS
VOL
0
ISOLATED 1 → 0 TRANSITION
ISOLATED 0 → 1 TRANSITION
ISOLATED 1 0 1
ISOLATED 0 1 0
SUPER IMPOSING ALL THE ABOVE SIGNALS
BINARY EYE PATTERNS
1 BIT TIME
1 BIT TIME

NRZ DATA — BINARY EYE PATTERN


UNIT INTERVAL tUI EQUALS ONE BIT TIME
OPTIMUM RECEIVER THRESHOLD LEVEL
EARLIEST THRESHOLD CROSSING tETC
LATEST THRESHOLD CROSSING tLTC

$$\text{PEAK-TO-PEAK JITTER} = \frac{t_{ETC} - t_{LTC}}{t_{UI}} \times 100\%$$

Since the shape of the waveform displayed by the oscilloscope resembles an eye, the pattern is called a "binary eye pattern".

The binary eye pattern provides a very useful tool to measure data signal quality. The spreading of the pattern at the receiver's threshold level indicates the peak-to-peak time jitter of the signal due to intersymbol interference. If the receiver's threshold level (slicing point) is displaced from its optimum value (at the mean of the signal swing produced by the driver), then the data recovered by the line receiver will incur additional time distortion over that already caused by the intersymbol interference. Displacement of the threshold level toward the One level will cause received Ones to be relatively shorter than the Zeros received. Likewise, a receiver threshold displacement toward the Zero level will cause received Zeros to be relatively shorter than the received Ones. This effect is called bias distortion, and is the result of offset (or bias) in either the receiver's threshold, or asymmetry in the driver output levels. The net effect of bias distortion is to cause the duration of one logic state to be shortened while the duration of the opposite logic state is lengthened. To reduce bias distortion to a minimum, the driver must produce opposite signal levels with good matching, and the receiver should slice at exactly the mean of the signal swing of the driver.

The height of the open space in the eye pattern directly gives the measure of the noise margin of the system. When no clear transition free area in the eye pattern exists, the eye is termed "closed", which indicates that error free data transmission is not possible at the data rate and line length without resorting to equalizing techniques. In some extreme cases, error free data will not be possible even with equalizing.

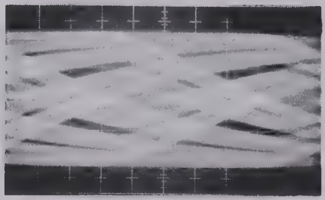
By using the eye pattern to indicate signal quality at the end of a line, a graph can be constructed for a particular cable showing the tradeoffs in signal quality (peak-to-peak jitter expected) as a function of line length and data rate for a given pulse code (NRZ, Polar RZ, BiΦ-L, etc.). An example graph for NRZ data is shown in the figure. The graph was constructed using data from eye pattern measurements on a 24 AWG twisted pair line driven by a voltage source type driver and parallel terminated in its characteristic resistance. The graph is representative of the performance available with ordinary twisted pair cables using NRZ data. Since coaxial cables have a much wider bandwidth at a given cable length, their signal quality curves would be shifted toward the right (i.e., higher maximum data rate at a given line length with a specified amount of peak-to-peak jitter allowed).

TRANSMISSION LINE INTERFACE ELEMENT APPLICATIONS

DATA SIGNAL QUALITY

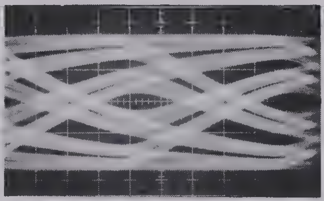
TYPICAL NRZ DATA EYE PATTERNS

(PLATE 1)



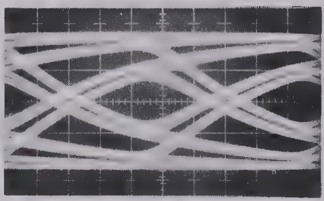
100% JITTER
EYE IS CLOSED —
ERROR FREE RECOVERY
OF NRZ DATA
PROBABLY NOT POSSIBLE

(PLATE 2)



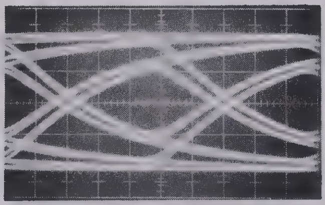
50% JITTER

(PLATE 3)



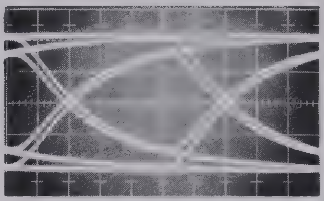
30% JITTER

(PLATE 4)



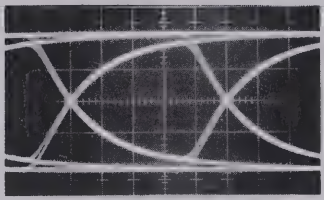
20% JITTER

(PLATE 5)



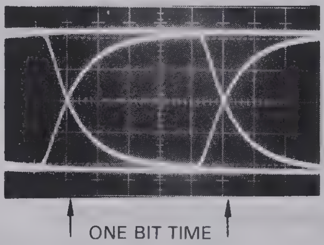
10% JITTER

(PLATE 6)



5% JITTER

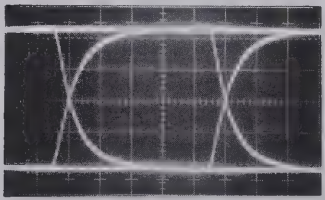
(PLATE 7)



NO
INTERSYMBOL
INTERFERENCE
$T_{UI} = 2\,T_r$

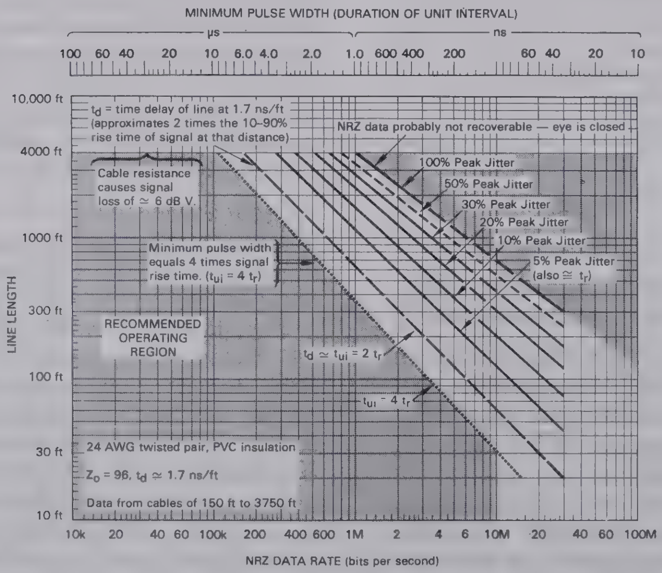
ONE BIT TIME
(ONE UNIT INTERVAL)

(PLATE 8)



NO
INTERSYMBOL
INTERFERENCE
$T_{UI} = 4\,T_r$

SIGNAL QUALITY AS A FUNCTION OF LINE LENGTH AND DATA RATE



t_d = time delay of line at 1.7 ns/ft (approximates 2 times the 10–90% rise time of signal at that distance)

NRZ data probably not recoverable — eye is closed

Cable resistance causes signal loss of ≃ 6 dB V.

Minimum pulse width equals 4 times signal rise time. ($t_{ui} = 4\,t_r$)

24 AWG twisted pair, PVC insulation

$Z_o = 96$, $t_d \simeq 1.7$ ns/ft

Data from cables of 150 ft to 3750 ft

The oscilloscope photographs show the eye patterns for NRZ data with various percentages of peak-to-peak jitter. As a general guideline, data communications circuits should be designed such that the time for one data bit is greater than four times the signal rise time at the end of the line. This almost always guarantees that the peak-to-peak time jitter of the recovered data due to intersymbol interference is less than 5% of the nominal bit time. This also minimizes the effect of any bias distortion introduced by driver offset or receiver bias since the signal slope at the nominal threshold crossing is relatively steep. A moderate change in threshold point causes only a small amount of bias distortion.

The signal quality graph shows the "best case" or minimum jitter expected at a given line length, and data rate for a particular cable, and only when the following conditions are met:

- Driver One and Zero levels are matched exactly
- Receiver threshold is exactly the mean of the One and Zero levels produced by the driver
- Time delays through both driver and receiver for both logic states are symmetrical and have zero skew (important when the minimum bit duration is close to the propagation delay of the devices)
- The line is perfectly terminated (no reflections on the circuit)

If any of the above conditions are not satisfied, the signal quality is less than predicted by the graph. Thus, the bit duration should be greater than four times the signal rise and fall time at the end of the line to allow for tolerances inherent in IC drivers and receivers.

If the line is operated unterminated with a voltage source type driver, the signal quality curves shown should be moved to the left (towards shorter lengths at a given data rate). Experiments have shown the required length reduction factor to be approximately 1/3. That is, an unterminated line of 333 feet shows about the same amount of intersymbol interference as a terminated line of 1000 feet at a given data rate with all other conditions the same.

If the desired NRZ data rate and line length fall between the $T_{UI} = 4\,T_r$ and the non-recovery line on the signal quality graph, eye pattern measurements should be made on the driver and cable selected for system usage. Direct eye pattern measurement is also recommended whenever a pulse code other than NRZ is selected for system usage.

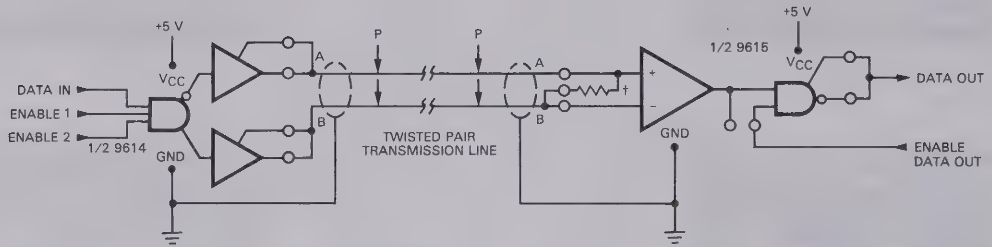
SELECTING LINE DRIVERS AND LINE RECEIVERS

The problem of selecting which particular line drivers and receivers to use depends more on the desired system characteristics, rather than merely the device electrical characteristics. The ultimate performance of the completed system will depend greatly on the form and mode of operation selected for the data communications circuits, and on the line drivers and line receivers themselves. Subtle peculiarities inherent in the form and mode of the data communications circuit selected may prevent the final system from achieving the design goals. Data communications circuit design requires the designer to keep in mind the tradeoffs between external noise and noise immunity, signal quality and line length, serial or parallel structure, and cost versus performance. The previous sections in this chapter have discussed the major design considerations involved in designing a data communications circuit. Some additional points follow.

- The most limiting factor to data rate with long lines (> 50 feet) is usually the rise and fall time of the cable. The use of the eye pattern will allow easy measurement of signal quality (amount of time jitter).
- Balanced differential forms are preferable to single ended forms where high noise environments are present. Opto-isolators and transformers offer high common mode operating ranges and ground isolation. With transformers, a dc free, self clocking code such as Polar RZ or BiΦ (digital binary phase modulation) is useful.
- The total number of ports on a multiplex system should be restricted so that the parallel combination of the input impedances of receivers and the output impedances of disabled drivers is greater than the characteristic resistance of the transmission line.
- Ground returns are necessary for proper operation of integrated circuit line drivers and receivers. This may be accomplished by connecting the shield of the line to the ground pins of ICs connected to the line.
- Data rates above 10M bps will usually require ECL to be used instead of TTL types for the drivers and receivers, because of the 20 to 50 ns propagation delays in the TTL compatible devices.
- Liberal use of .01 to .1 µF capacitors to decouple the power supplies feeding line drivers and receivers is recommended. One capacitor per power supply for every two to four devices is usually sufficient.

TRANSMISSION LINE INTERFACE ELEMENT APPLICATIONS

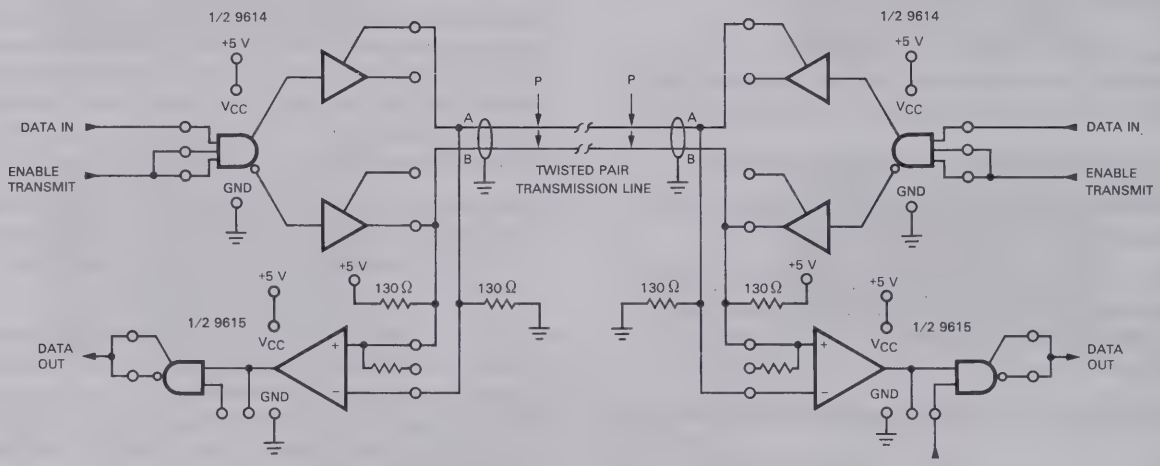
9614/9615 SIMPLEX, BALANCED DIFFERENTIAL


+5 V
VCC
DATA IN
ENABLE 1
ENABLE 2
1/2 9614
GND
A
B
P
P
TWISTED PAIR
TRANSMISSION LINE
A
B
†
+
−
GND
1/2 9615
+5 V
VCC
DATA OUT
ENABLE
DATA OUT

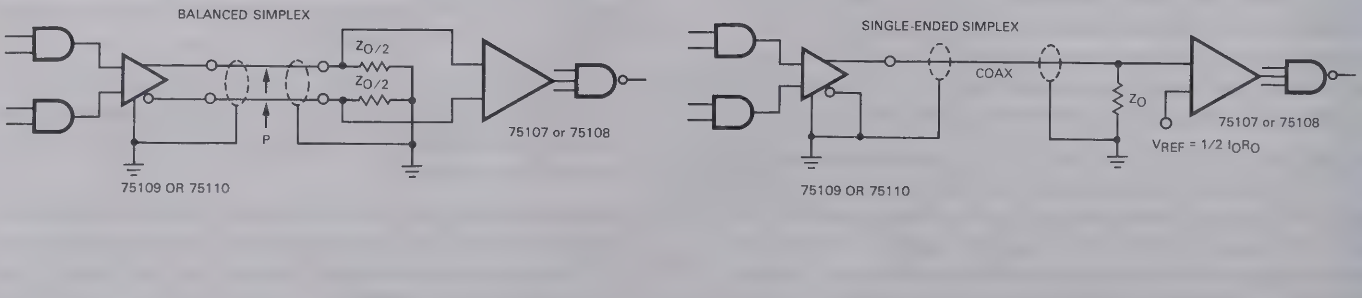
IF TRANSMISSION LINE IS SHIELDED, TIE SHIELD TO GROUND AT BOTH DRIVER AND RECEIVER ENDS OF LINE.

†INTERNAL TERMINATION RESISTOR MAY BE USED IF LINE IMPEDANCE ≃ 130 Ω
FOR PRECISE MATCHING, USE EXTERNAL TERMINATION RESISTOR.

9614/9615 HALF-DUPLEX, DIFFERENTIAL

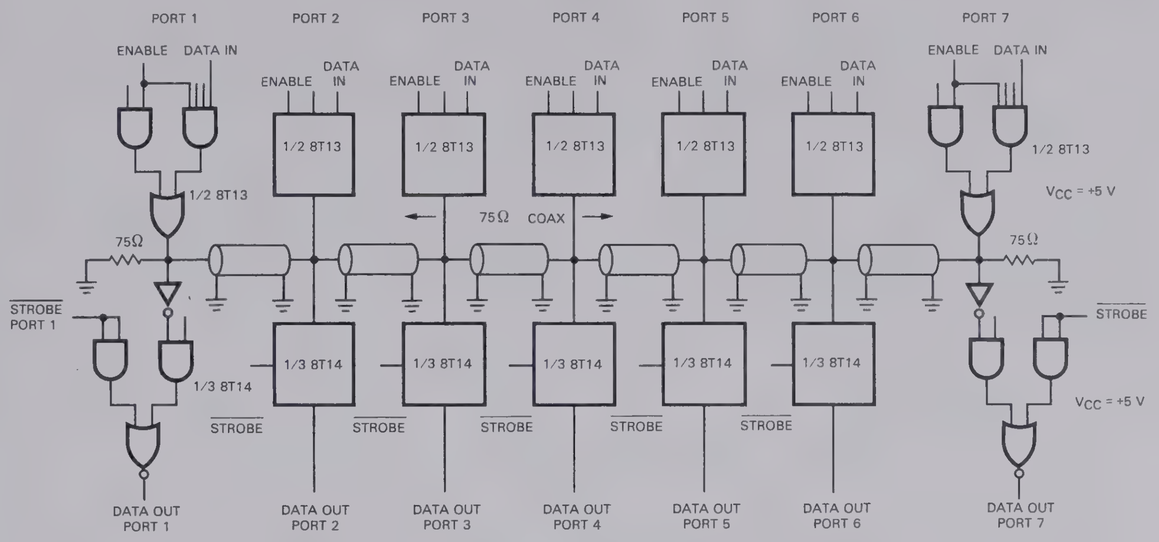

1/2 9614
+5 V
VCC
DATA IN
ENABLE
TRANSMIT
GND
P
P
A
B
TWISTED PAIR
TRANSMISSION LINE
A
B
1/2 9614
+5 V
VCC
DATA IN
ENABLE
TRANSMIT
GND
+5 V
+5 V
130 Ω
130 Ω
130 Ω
130 Ω
+5 V
+5 V
1/2 9615
VCC
GND
DATA
OUT
+
−
1/2 9615
VCC
GND
DATA
OUT
+
−

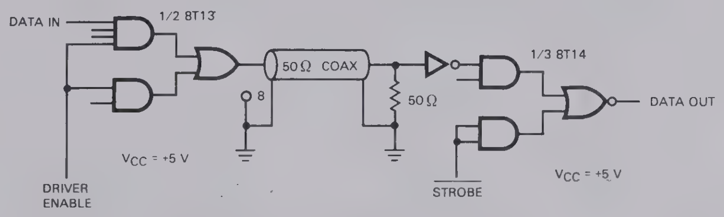
SIMPLEX MODE 75107 SERIES


BALANCED SIMPLEX
ZO/2
ZO/2
P
75107 or 75108
75109 OR 75110
SINGLE-ENDED SIMPLEX
COAX
ZO
75107 or 75108
VREF = 1/2 IORO
75109 OR 75110

TRANSMISSION LINE INTERFACE ELEMENT APPLICATIONS

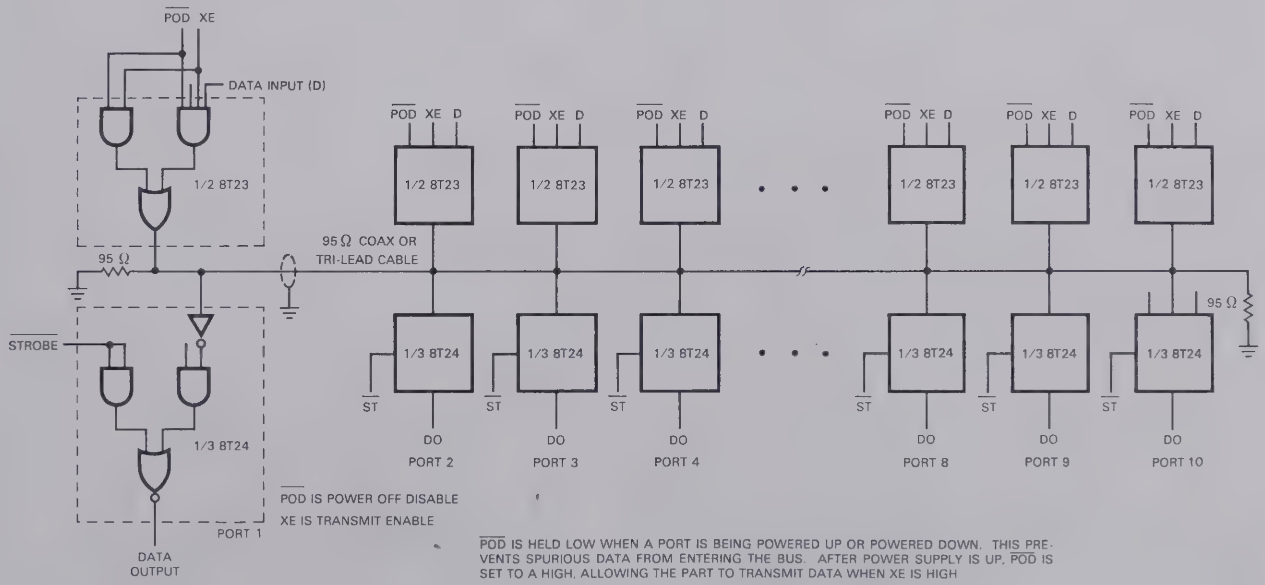
SINGLE ENDED — SIMPLEX OPERATION


PORT 1
PORT 2
PORT 3
PORT 4
PORT 5
PORT 6
PORT 7
ENABLE
DATA IN
1/2 8T13
1/3 8T14
75Ω
COAX
VCC = +5 V
STROBE
STROBE PORT 1
DATA OUT PORT 1
DATA OUT PORT 2
DATA OUT PORT 3
DATA OUT PORT 4
DATA OUT PORT 5
DATA OUT PORT 6
DATA OUT PORT 7


DATA IN
1/2 8T13
50Ω COAX
8
50Ω
1/3 8T14
DATA OUT
VCC = +5 V
DRIVER ENABLE
STROBE
VCC = +5 V

SINGLE ENDED SIMPLEX TRANSMISSION

IBM 360 I/O INTERFACE APPLICATION


POD
XE
DATA INPUT (D)
1/2 8T23
95 Ω
95 Ω COAX OR TRI-LEAD CABLE
STROBE
1/3 8T24
PORT 1
DATA OUTPUT
POD XE D
1/2 8T23
1/3 8T24
ST
DO
PORT 2
PORT 3
PORT 4
PORT 8
PORT 9
PORT 10
95 Ω

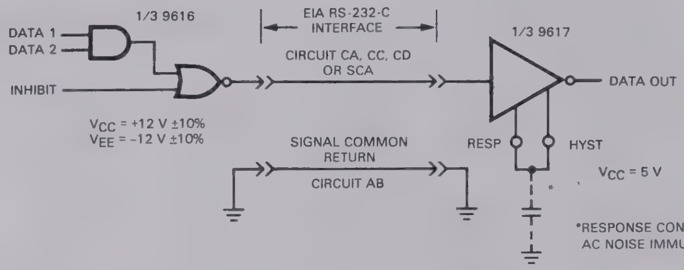
$\overline{\text{POD}}$ IS POWER OFF DISABLE
XE IS TRANSMIT ENABLE

$\overline{\text{POD}}$ IS HELD LOW WHEN A PORT IS BEING POWERED UP OR POWERED DOWN. THIS PREVENTS SPURIOUS DATA FROM ENTERING THE BUS. AFTER POWER SUPPLY IS UP, $\overline{\text{POD}}$ IS SET TO A HIGH, ALLOWING THE PART TO TRANSMIT DATA WHEN XE IS HIGH

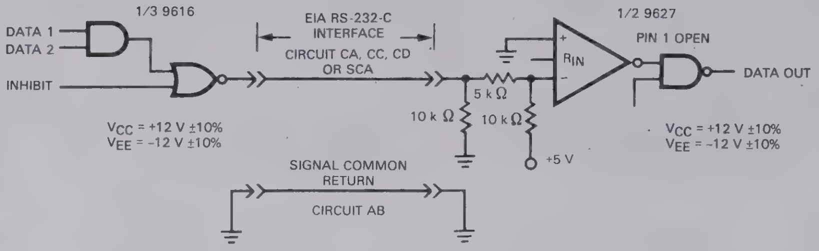
ONE BIT SLICE OF I/O BUS IS SHOWN

TRANSMISSION LINE INTERFACE ELEMENT APPLICATIONS

EIA RS 232-C INTERFACE WITH FAILSAFE RECEIVERS


1/3 9616
DATA 1
DATA 2
INHIBIT
V_{CC} = +12 V ±10%
V_{EE} = −12 V ±10%
EIA RS-232-C INTERFACE
CIRCUIT CA, CC, CD OR SCA
SIGNAL COMMON RETURN
CIRCUIT AB
1/3 9617
DATA OUT
RESP
HYST
V_{CC} = 5 V
*

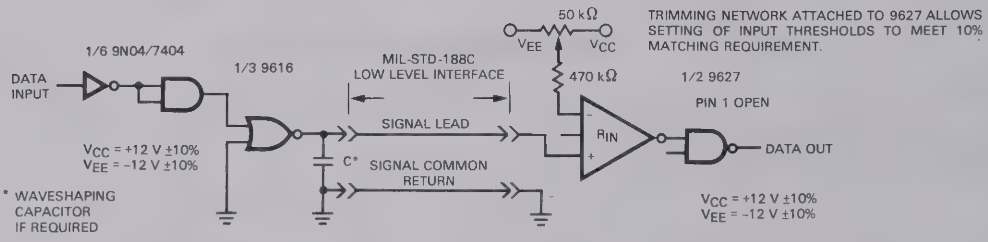
*RESPONSE CONTROL OPTIONAL TO INCREASE AC NOISE IMMUNITY OF RECEIVER


1/3 9616
DATA 1
DATA 2
INHIBIT
V_{CC} = +12 V ±10%
V_{EE} = −12 V ±10%
EIA RS-232-C INTERFACE
CIRCUIT CA, CC, CD OR SCA
SIGNAL COMMON RETURN
CIRCUIT AB
5 kΩ
10 kΩ
10 kΩ
+5 V
1/2 9627
PIN 1 OPEN
+
R_{IN}
−
DATA OUT
V_{CC} = +12 V ±10%
V_{EE} = −12 V ±10%

BOTH CIRCUITS SHOWN ABOVE PROVIDE FOR EIA FAILSAFE OPERATION. THE DATA OUTPUTS WILL BE LOW IF RECEIVER INPUTS ARE OPEN, OR SHORTED TO CIRCUIT AB, OR IF THE DRIVER IS POWERED-DOWN. THE DATA OUTPUTS WILL BE HIGH IF A SPACE (LOGIC "0" OR "ON") IS RECEIVED.

IF THE INHIBIT LEAD IS H, THE 9616 WILL PRODUCE A MARK LEVEL OUTPUT. INHIBIT L ALLOWS 9616 OUTPUT TO BE CONTROLLED BY DATA INPUTS. NO EXTERNAL CAPACITOR IS REQUIRED BY THE 9616 TO MEET SLEW RATE REQUIREMENTS OF EIA RS-232; THE 9616 HAS INTERNAL SLEW RATE CONTROL.

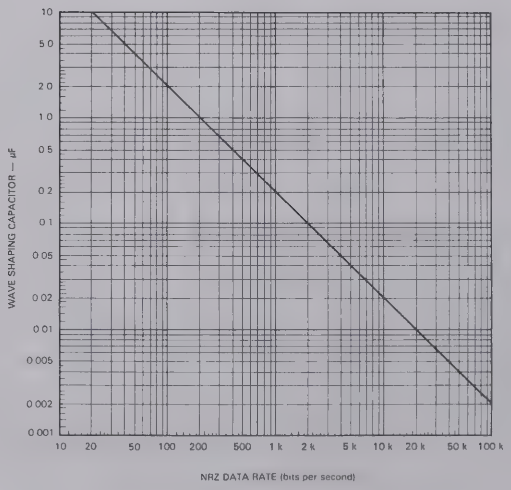
MIL STD-188C INTERFACE

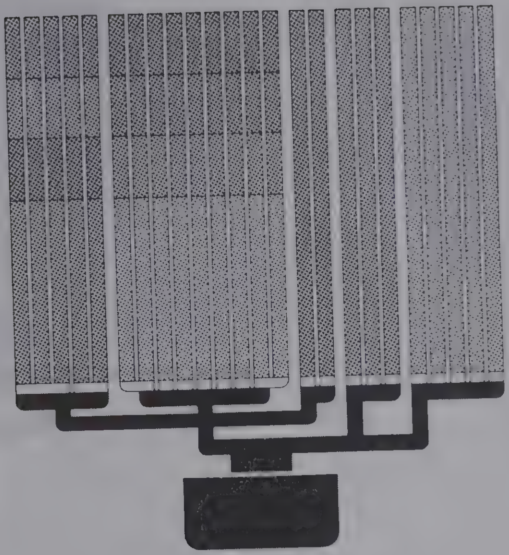

50 kΩ
V_{EE}
V_{CC}
470 kΩ
1/6 9N04/7404
DATA INPUT
1/3 9616
V_{CC} = +12 V ±10%
V_{EE} = −12 V ±10%
MIL-STD-188C LOW LEVEL INTERFACE
SIGNAL LEAD
C*
SIGNAL COMMON RETURN
1/2 9627
PIN 1 OPEN
−
R_{IN}
+
DATA OUT
V_{CC} = +12 V ±10%
V_{EE} = −12 V ±10%

TRIMMING NETWORK ATTACHED TO 9627 ALLOWS SETTING OF INPUT THRESHOLDS TO MEET 10% MATCHING REQUIREMENT.

* WAVESHAPING CAPACITOR IF REQUIRED

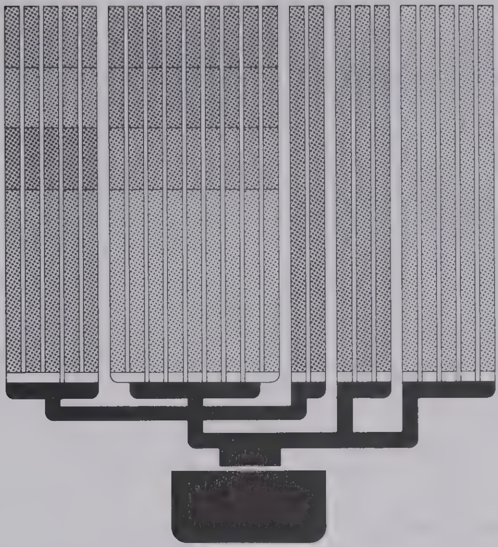
TYPICAL CAPACITOR VALUE REQUIRED VS NRZ DATA RATE





TTL Characteristics

Electrical Properties
Physical System Considerations
Flip-Flops and Registers
Interfacing



TTL CHARACTERISTICS

INTRODUCTION

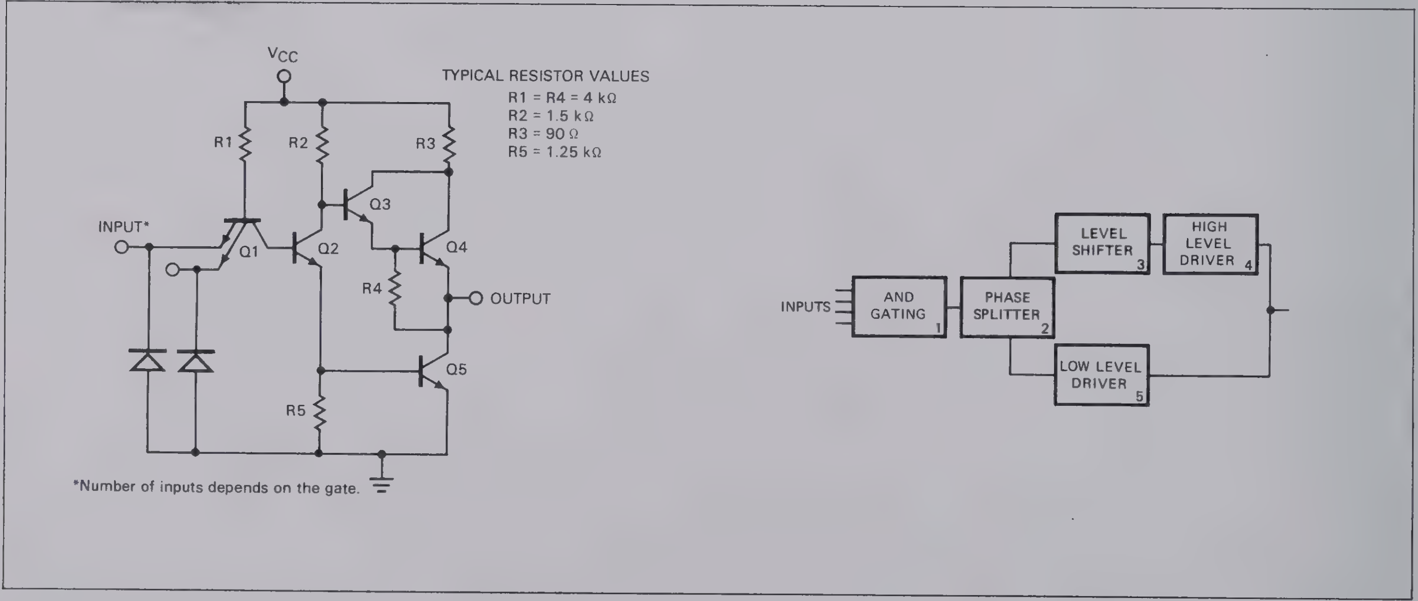
The previous chapters explain the logic operation of Fairchild TTL devices of medium and small scale complexity and their use in many different subsystems. The devices are described as functional units (black boxes) without special attention to electrical characteristics.

The following pages describe the electrical aspects of TTL components and systems. The first section describes the input/output characteristics, thermal characteristics and noise margins. The second presents physical system considerations when TTL devices are interconnected. It covers transmission line drive capability, ringing, crosstalk and gives general rules for circuit layout. The third section discusses the timing aspects of flip-flops and registers and explains the clock skew problem and how to avoid it. The last section shows how TTL devices can interface with other integrated circuit families and transistors.

INTRODUCTION

The TTL devices described in this book differ widely in function and complexity, but their electrical input and output characteristics are very similar and are defined, tested, and guaranteed in common terms. As a result, all members of the TTL family interface perfectly. Circuit characteristics can be easily described with a 2-input NAND gate example.

NAND Gate Example


V_{CC}
TYPICAL RESISTOR VALUES
R1 = R4 = 4 kΩ
R2 = 1.5 kΩ
R3 = 90 Ω
R5 = 1.25 kΩ
R1
R2
R3
Q3
INPUT*
Q1
Q2
Q4
R4
OUTPUT
Q5
R5
*Number of inputs depends on the gate.
INPUTS
AND GATING 1
PHASE SPLITTER 2
LEVEL SHIFTER 3
HIGH LEVEL DRIVER 4
LOW LEVEL DRIVER 5

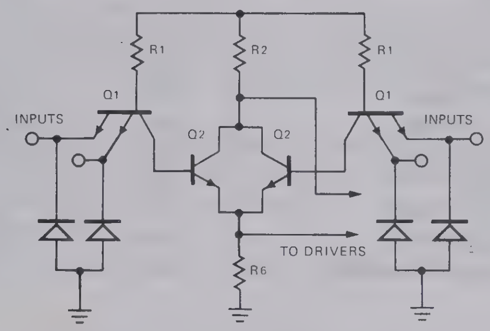
CIRCUIT DESCRIPTION

The circuit is actually five subcircuits and each performs a separate function. The input circuit is a multi-emitter transistor functioning as an AND gate. When all inputs are High (<2 V) the current in R1 ($\approx$1.2 mA) flows through the collector of Q1 into the base of Q2, turning on Q2 which then turns on Q5, generating a Low output voltage. The base of Q1 is clamped three diode drops ($2\ V_{BE} + V_{CB}$) above ground, $\approx$2 V at room temperature. If any input goes Low (<0.8 V), the current in R1 flows through the emitter of Q1, out of the input lead into ground. Any stored charge in the base of Q2 is quickly removed through the transistor action of Q1, and Q2 is thus rapidly turned off. This turns off Q5 and turns on Q3 and Q4, resulting in a High output voltage.

The multi-emitter input transistor Q1 has an input leakage current that is significantly higher than that of DTL. In TTL circuit design, this input leakage current is limited by reducing the inverse current gain of the input transistor through proper choice of transistor geometry and processing.

The phase-splitter transistor Q2 produces complementary drive signals for Q3 and Q5. The collector and emitter of the phase splitter can be connected to additional phase-splitting transistors driven by AND gates, thus providing the logic AND/OR/INVERT function. This mechanism is a very powerful logic tool, enabling complex logic functions to be realized in a minimum integrated circuit chip area with a minimum number of gate delays.

AND/OR/INVERT Function



The Darlington connected transistors Q3 and Q4 act as a low impedance High level driver, speeding up the Low to High output transition and providing superior ac drive capability and good noise immunity in the High state. Resistor R4 acts as a current limiter for ac switching currents and for accidental output short circuits to ground. Resistor R5 provides a base turn-off current path for transistor Q4. Connecting R5 to the output rather than to ground reduces power consumption. Q5 is the Low level driver and R6 provides base turn-off current to ground.

OUTPUTS

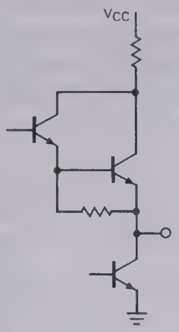
Different TTL devices use variations on the output pull up structure. Each configuration has advantages and disadvantages, both from the manufacturer's and the user's point of view, as outlined in the comparison chart.

A resistive pull up can be added to any TTL output circuit, increasing V_{OH} to almost V_{CC}, but only circuits *c, d,* and *e* can be pulled higher than V_{CC}, e.g., to +7 V for driving MOS circuits. Configurations *a* and *b* have a diode associated with the resistor at the output which clamps the output one diode drop above V_{CC}. This is an important consideration in large systems where sections might be powered down ($V_{CC} = 0$). In this state, the outputs of circuits *a* and *b* represent a very low impedance at a fairly low voltage (<1 V); while the outputs of circuits *c, d,* and *e* represent a high impedance and thus a logic High, more appropriate for isolation from the rest of the system.

Most modern designs use a more sophisticated active pull down structure in the base of Q5. This improves the transfer characteristic, increases noise immunity and speeds up the output delay.

TTL Output Configurations

a. DARLINGTON



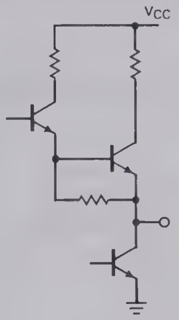
ADVANTAGE:

High ac drive capability
$V_{OH} = V_{CC} - V_{BE}$ at $I_O = 0$
Small size (transistors share one common isolation)

DISADVANTAGE:

Output cannot be pulled higher than one diode drop above V_{CC}

b. 2-STAGE EMITTER FOLLOWER ("DARLINGTON SPLIT")



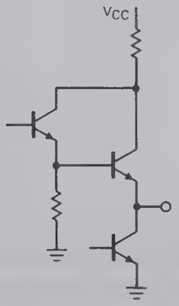
ADVANTAGE:

High ac drive capability
$V_{OH} = V_{CC} - V_{BE}$ at $I_O = 0$

DISADVANTAGE:

Larger than circuit A
Output cannot be pulled higher than one diode drop above V_{CC}

c. DARLINGTON WITH RESISTOR TO GROUND



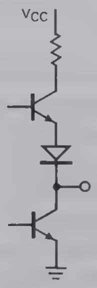
ADVANTAGE:

High ac drive capability
Lower $V_{OH} = (V_{CC} - 2\ V_{BE})$ increases speed
Outputs can be pulled higher than V_{CC}

DISADVANTAGE:

Higher dissipation
Lower noise immunity in the High state

d. TRANSISTOR — DIODE



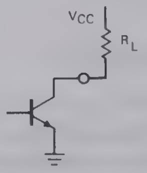
ADVANTAGE:

Lowest power consumption
Small size
Outputs can be pulled higher than V_{CC}

DISADVANTAGE:

Less ac drive capability

e. OPEN COLLECTOR



ADVANTAGE:

Bussable, allows collector ANDing (Wired-OR)

DISADVANTAGE:

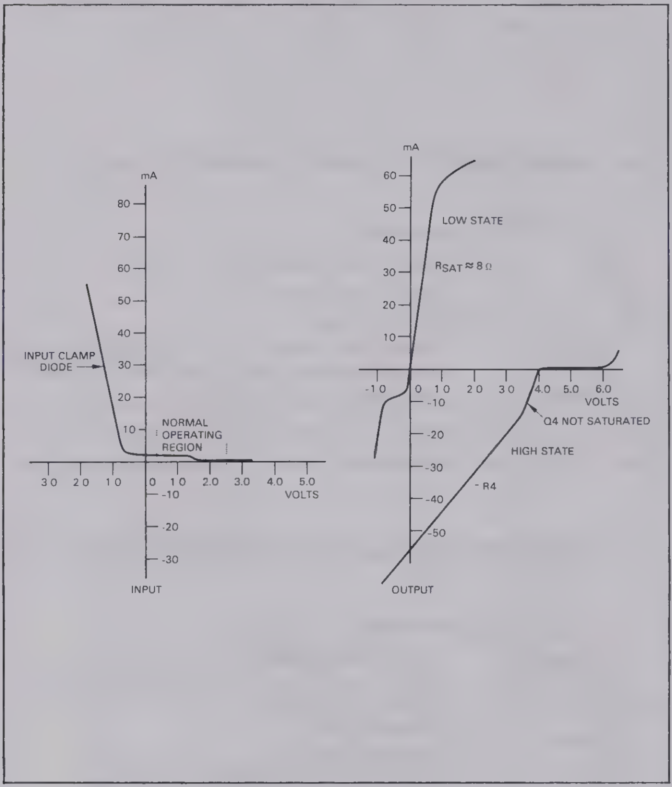
High output impedance in the High state
Slow, especially with capacitive loading
Requires additional resistor

TTL CHARACTERISTICS • ELECTRICAL PROPERTIES

TTL INPUT AND OUTPUT IMPEDANCE

The input impedance of a TTL circuit is very high for positive input voltages > 2 V; it is ≈ 4 kΩ between +1 and −0.5 V and is very low for more negative input voltages. Each TTL input has a clamp diode to ground that ensures this low input impedance for negative voltages. These clamp diodes protect the TTL input and limit any negative input swing due to inductive ringing or reflections on transmission line interconnections.

Typical Input/Output Characteristics



The output impedance of a typical TTL device in both the Low and High state is shown above. In the Low state, the output impedance is determined by a saturated transistor (≈ 8 Ω). However, at very high sinking current, especially at low temperature, the output device is not able to stay in saturation and the output impedance rises as shown.

When switching from the Low to the High state, the totem pole output structure provides a low output impedance capable of rapidly charging capacitive loads. However, charge and discharge currents must also flow through the V_{CC} and ground distribution networks. The V_{CC} and ground lines should therefore be short and adequately decoupled. Moreover, if during the Low to High transition, transistor Q5 has not turned off by the time Q4 is turned on, there is a narrow current spike through the totem pole, which acts as a noise generator unless the supply is properly decoupled.

SUPPLY VOLTAGE AND TEMPERATURE RANGE

The nominal supply voltage V_{CC} for all TTL circuits is +5 V. Commercial grade parts are guaranteed to perform with a ±5% supply tolerance (±250 mV) over an ambient temperature range of 0 to 75°C (some to 70°C). Mil grade parts are guaranteed to perform with a ±10% supply tolerance (±500 mV) over an ambient temperature range of −55 to 125°C.

The actual junction temperature can be calculated by multiplying the power dissipation of the device with the thermal resistance of the package and adding it to the measured ambient temperature T_A or package (case) temperature T_C.

Below are listed some of the standard dual in-line packages (DIP) and Flatpaks used by Fairchild, including typical junction-to-ambient thermal resistance θ_{JA} and typical junction to-case thermal resistance θ_{JC}.

PACKAGE	θJA °C/W	θJC °C/W
16-lead Flatpak	117	11
24-lead Flatpak	91	9
14-lead DIP, SSI	119	33
16-lead DIP, SSI	105	25
24-lead DIP	53	7
14-lead DIP, MSI	115	30
16-lead DIP, MSI	97	22
14/16-lead DIP, plastic	125	20

Example: A 9301 in ceramic DIP dissipates typically 145 mW. At 55°C ambient temperature the junction temperature is:

$$T_J = (0.145 \times 97) + 55 = 69°C$$

INPUT AND OUTPUT VOLTAGES

The following input and output voltage levels are guaranteed over a worst-case combination of the appropriate temperature and supply voltage range and, for the output, over a specified current range:

	Standard & High Speed TTL	Low Power TTL	Schottky TTL
Maximum Low level output voltage V_{OL} max	0.4	0.3	0.5 V
Minimum High level output voltage V_{OH} min	2.4	2.4	2.5 V
Maximum Low level input voltage V_{IL} max	0.8	0.7	0.8 V
Minimum High level input voltage V_{IH} min	2.0	2.0	2.0 V

V_{OL} and V_{OH} are the voltages generated by the output. V_{IL} and V_{IH} are the voltages required at the input to generate the appropriate output levels. (See discussion of dc noise immunity below). The numbers given above are guaranteed worst-case values. Typical values are given in the data sheets.

V_{OL} is the sum of <100 mV off set voltage and a resistive drop that, within the limits of the output sink capability, is proportionate to the sink current. If only half the specified maximum fan out is used, V_{OL} for a standard gate will be below 250 mV. The inherent off set voltage of Schottky outputs is higher, ≈ 200 mV.

V_{OH} depends on the output circuit configuration. The Darlington configuration with a 1 kΩ resistor to output has the following characteristics:

When sourcing negligible current, i.e., under normal stationary operating condition, Q4 is not conducting. V_{OH} is one diode drop below V_{CC} with a temperature coefficient of -2 mV/°C and an impedance of 1 kΩ.

When sourcing moderate current (a few mA), the V_{OH} is two diode drops below V_{CC} with a temperature coefficient of −4 mV/°C and a very low impedance of $< 30\,\Omega$.

When sourcing large current ($\approx$20 mA), the Darlington configuration is saturated and the output follows the resistive load line determined by the current limiting resistor R4.

V_{IL} and V_{IH} have a temperature coefficient of −4 mV/°C.

INPUT AND OUTPUT CURRENTS

Four different current specifications are needed to characterize the normal operation of a TTL device:

Input Low Current	I_{IL}	the current flowing out of the input when it is held in the Low state (0.4 V).
Input High Current	I_{IH}	the current flowing into the input when it is held High. (leakage or reverse Beta current).
Output Low Current	I_{OL}	the current that the output can sink without exceeding the specified output Low voltage V_{OL}.
Output High Current	I_{OH}	the current that the output can source without dropping below the specified output High voltage V_{OH}.

To simplify interconnection rules, these currents have been normalized as TTL Unit Loads (UL). One UL is equivalent to the worst-case input current of a standard TTL gate, 1.6 mA in the Low state and 40 µA in the High state. The input requirements of every TTL circuit are given in Unit Loads. Some MSI inputs have a fan in of several UL, high speed and Schottky circuits have a fan in of 1.25 UL, while low power TTL has a fan in of only 0.25 UL. The drive capability of the TTL output is expressed in the same terms. The specified maximum fan out is the number of Unit Loads that the output can drive under worst case conditions.

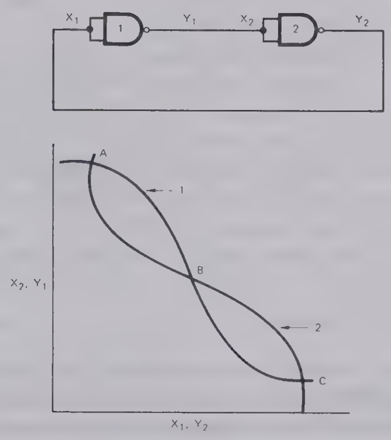
Modern TTL outputs use a 2-transistor pull up structure, which improves ac performance and also provides twice the high drive capability compared to the low sink capability. Unused inputs can therefore be tied to used inputs of the same NAND gate without burdening the fan out capability of the driver. When several inputs of one NAND gate are interconnected, the input High currents are added, but the input Low current is the same as for one input.

DC NOISE MARGINS

Knowledge of noise margins in digital devices is crucial to the systems engineer in specifying signal and power transmission requirements. This knowledge also is essential in deciding whether shielding is needed to lower external noise. How noise margins occur and what range of quantities represent the levels in a digital system are best explained by studying the transfer function of the device.

Consider the simple feedback system built with two identical devices. The transfer characteristic of device 2 is drawn on the same diagram with device 1, with the input forcing function X_2 on the vertical scale and the resulting response Y_2 on the horizontal scale.

Simple Feedback System and Transfer Function



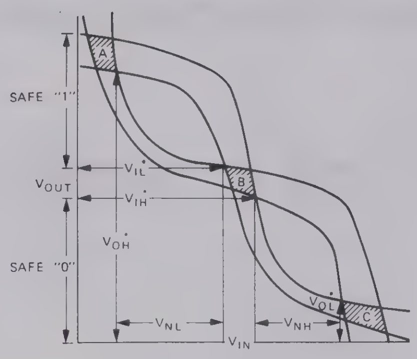
The characteristics cross at three points and since the ordinate represents X_2, Y_1 and the abscissa X_1, Y_2 these three points represent the levels at which the simple digital feedback system can reside: only at these three points is $X_2 = Y_1$, and $X_1 = Y_2$. If device 1 has input and output quantities specified by point A, then the input and output of device 2 must be specified by point A also. Both devices can have identical inputs and outputs, represented by point B, but this is an unstable condition.

In practice, the devices in the system would not be identical. The transfer characteristic, therefore, would not be a single line but would lie within an envelope. This envelope would contain the characteristic of any device used in the system.

The dimensions of the envelope are determined by the output limits accepted by testing. An arbitrary point, called V_{OL} is selected as the highest output voltage acceptable as a logic 'zero'. A guard band is added to this, and the V_{OL} plus guard band is applied to an input. The resulting output voltage is designed as V_{OH}, the lowest acceptable logic 'one' voltage. These parameters, V_{OL} and V_{OH} are used to sort devices, and the envolope of transfer characteristics result. Two of these envelopes are plotted together below. Now the device levels may reside in one of the three areas of intersection. Again, the center area is unstable.

The point at the upper left hand corner of region C represents the highest low output voltage a device can have if driven by another device in the simple feedback system. This output voltage is designated V_{OL}^*. A device driven by V_{OL}^* will have the output voltage at the lower right corner of Region A. These levels are the outputs of interconnected devices at the worst case accepted in sorting.

Noise Margins/Characteristic Envelopes



An input level, however, may extend all the way up to the unstable region B and still be interpreted as a logic Low. The output voltage falls to the opposite corner of region B. If this happens, a change of state of the system does not occur. These points, on the corners of region B, represent the most extreme input levels which do not result in a change in the system. They are designated V_{IL} * and V_{IH} *. The noise margin of the system is the vertical separation between the outputs in a noiseless system and the inputs limits V_{IL} * and V_{IH} *. These differences are V_{NH} and V_{NL}, the high and low noise margins. Note that if the envelopes become wider due to looser testing limits, or if the slope of the transition region becomes less steep, region B elongates toward regions A and C, resulting in a decreased noise margin.

In the data sheets, the parameters given are V_{OH}, V_{OL}, V_{IH}, and V_{IL}. These are not the starred quantities on the diagram. The numbers in the data sheets are determined as follows:

V_{OL}	Arbitrary
V_{IL}	V_{OL} + Guard band V_{NL}
V_{OH}	Output when input is V_{IL}
V_{IH}	V_{OH} – Guard band V_{NL}

The apparent noise margins may then be determined by subtracting V_{OL} from V_{IL} and V_{IH} from V_{OH}. The relationships between these points and the starred quantities, which give the real noise margins, are as follows:

V_{IL} =	V_{IL}^*
V_{IH} =	V_{IH}^*
V_{OH}	Lies on the transfer curve slightly to the right of and below region A.
V_{OL}	Lies on the curve slightly to the left of and above region C.

The actual noise margins then exceed the apparent margins given in the data sheets.

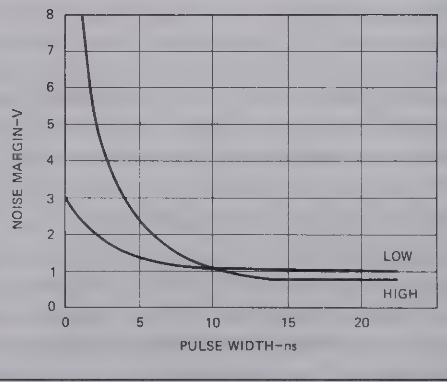
TTL AC NOISE MARGINS

In the discussion of dc noise margins, the transfer characteristics were assumed to be dependent only upon the amplitude of input voltage and not on its rate of change. However, a digital circuit cannot respond immediately to an input waveform and most devices exhibit hysteresis. The ac noise margin of a circuit may therefore be different from the dc margin. In a well designed digital circuit with a low pulse rate and large pulse width, the ac noise margin should not be detectably different from the dc noise margin. But, as the pulse width is narrowed, the noise margin changes depending on the coupling used. In certain types of circuits using capacitive coupling, the ac noise margin can decrease with decreasing pulse width. However, with the majority of circuit designs using dc coupling like the 9000 and 9300 series devices, noise margin increases with a decrease in pulse width.

This effect occurs at pulse widths less than the device propagation delay. It is caused by the inability of the circuit to respond instantaneously because of stored charge and external loads. Narrow noise spikes at very high frequency can have the effect of biasing a device. This can result in lowering the low frequency noise margin. Generally this biasing effect is small.

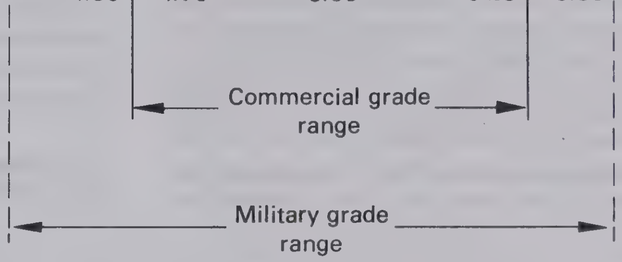
The main interest in ac noise margins is the values of pulse widths caused by the circuits themselves acting as noise generators. This noise is caused by logic spikes resulting from different path delays and by voltage and current changes in distributed impedances. The device-generated noise usually has a narrow pulse width and high frequency. The noise margins to this internally generated noise should be superior to the dc noise margin.

AC Noise Margins vs Pulse Width



TRANSMISSION LINE DRIVE CAPABILITY

TTL FAMILY OR DEVICE	COLLECTOR RESISTOR R	Lowest Transmission Line Impedance				
		WORST CASE (R + 30%)		NOMINAL	BEST CASE (R – 30%)	
9N/74 Series	130	241.4	204.8	136.8	84.6	75.8
9000 Series	80	148.5	126.0	84.2	52.0	46.6
9H/74H Series	58	107.7	91.3	61.0	37.7	33.8
9S/74S Series	55	110.0	92.2	61.1	37.5	33.4
9L Series	320	594.2	504.2	336.8	208.3	186.6
9009	50	92.8	78.7	52.6	32.5	29.1
9N40/7440	100	185.7	157.5	105.2	65.1	58.3
9H40/7440	60	111.4	94.5	63.1	39.0	35.0
9S40/7440 9S140/74S140	25	50.0	41.9	27.7	17.0	15.2
Supply Voltage (V_{CC})		4.50	4.75	5.00	5.25	5.50



Practical transmission lines, cables and strip lines used for TTL interconnections have a characteristic impedance between 50 and 150 Ω. Thus none of the standard or low power TTL circuits can drive a transmission line, and only the 9S40/9S140 is truly capable of driving a 50 Ω line under worst case conditions.

These considerations, applicable only when the round trip delay of the line is longer than the rise or fall time of the driving signal ($2t_d > t_r$), do not affect most TTL interconnections. Short interconnections do not behave like a resistive transmission line, but more like a capacitive load. Since the rise time of different TTL outputs is known, the longest interconnection that can be tolerated without causing transmission line effects can easily be calculated and is listed in the table below.

PC Board Interconnections

TTL FAMILY	RISE TIME	FALL TIME	MAX INTERCONNECTION LENGTH
9L, 93L	14–18 ns	4–6 ns	18 in. (45 cm)
9000, 9300 9N/74N	6–9 ns	4–6 ns	18 in. (45 cm)
9H/74H	4–6 ns	2–3 ns	9 in. (22.5 cm)
9S/74S 93S	1.8–2.8 ns	1.6–2.6 ns	7.5 in. (19 cm)

Assuming 1.7 ns/foot propagation speed, typical for epoxy fiberglass PC boards with ε_r = 4.7.

Slightly longer interconnections show minimal transmission line effects; the longer the interconnections, the greater the chance that system performance may be degraded due to reflections and ringing. The chapter on line driving and receiving gives additional information on transmission line phenomena on longer lines. Good system operation can generally be obtained by designing around 100 Ω lines. A 0.026 inch (0.65 mm) trace on an epoxy-glass board (ε_r = 4.7) with a ground plane on the other side represents a 100 Ω line. 28 to 30 gauge wire (0.25 to 0.30 mm) twisted together forms a twisted pair line with a characteristic impedance of 100 to 115 Ω. Wire over ground screen (3/4 squares) gives 150 to 250 Ω with a significant improvement in propagation speed, since the dielectric constant approaches that of air.

RINGING

By far the worst problem with high speed TTL circuits is the possible rebound (ringing) of the signal into the input threshold region (0.8—2.0 V) following a High-to-Low level change. When a driver switches from a High-to-Low state, the output voltage should fall below the threshold value. However, a line having a very low characteristic impedance does not allow transistor Q5 in the NAND gate example to saturate, and the resulting output voltage may not be low enough to switch an adjacent device until after two or more line delay times. A worst-case situation can occur at the input of the driven device at the end of the line. The voltage at the receiver can swing negative and then positive after reflection from the driver ($3t_d$ of the line). This positive excursion, which originates after three line delay times and lasts for two more line delay times, can rise above the threshold voltage causing the receiver to switch erroneously. This effect is most pronounced when the driver is fully loaded but driving only one distant device. A similar situation may occur when the driver is switching from the Low-to-High state, but in this case the output impedance of the driver is larger and the effect, if present, is less pronounced. The input diode on each input of a TTL device limits the negative swing at the input of an element, thus lowering the next positive transient. These input diodes are extremely effective in reducing the amount of noise and transients in a digital system built with high speed circuits.

INTRODUCTION

The properties of high speed logic gates discussed in the preceding section on electrical characteristics dictate that some care be used in design and layout of a system. In this section, some specific details on systems requirements are presented.

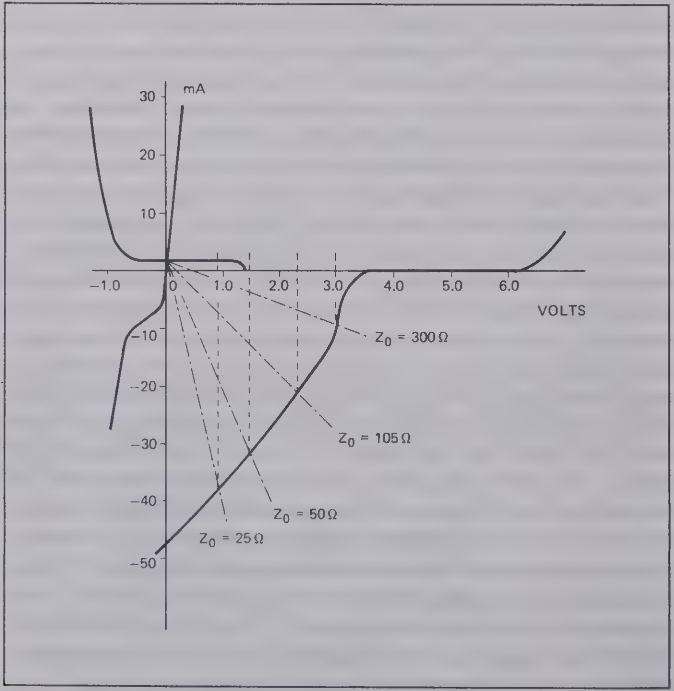
TRANSMISSION LINE EFFECTS

The fast rise and fall times of TTL outputs (2 to 6 ns) produce transmission line effects even with relatively short (< 2 ft) interconnections. Consider one TTL device driving another, and the driver switching from the Low to the High state. If the propagation delay of the interconnection is long compared to the rise time of the signal, the arrangement behaves like a transmission line driven by a generator with a non-linear output impedance. Simple transmission line theory shows that the initial voltage step at the output just after the driver has switched is

$$V_{OUT} = V_E \left(\frac{Z_O}{Z_O + R_O} \right)$$

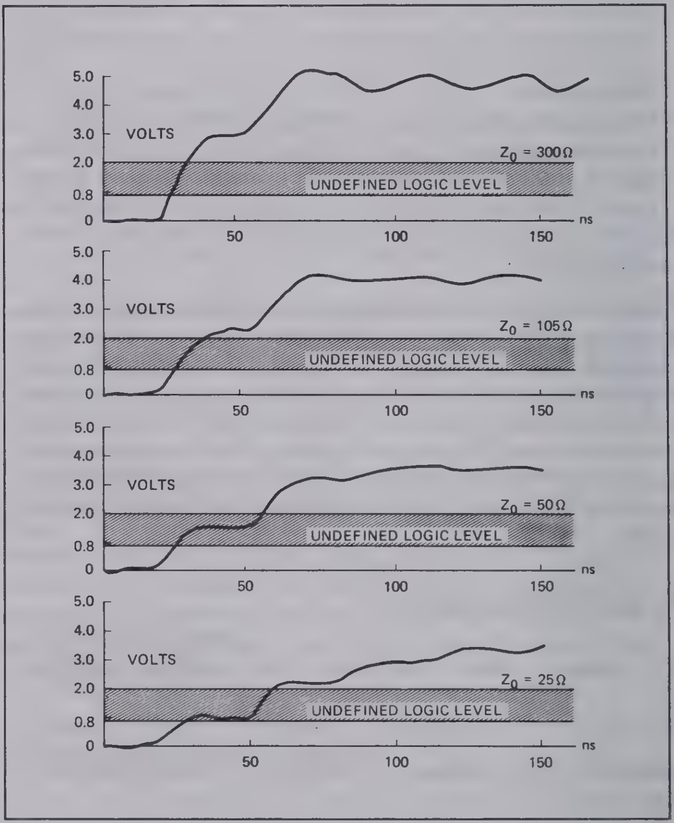
where Z_O is the characteristic impedance of the line, R_O is the output impedance of the driver, and V_E is the equivalent output voltage source in the driver, V_{CC} minus the forward drop of the pull-up transistors.

Initial Output Voltage of TTL Driving Transmission Line



This initial voltage step propagates down the line and reflects at the end, assuming the typical case where the line is open ended or terminated in an impedance greater than its characteristic impedance Z_O. Arriving back at the source, this reflected wave increases V_{OUT}. If the total round-trip delay is larger than the rise time of the driving signal, there is a staircase response at the driver output and anywhere along the line. If one of the loads (gate inputs) is connected to the line close to the driver, the initial output voltage V_{OUT} might not exceed V_{IH}. This input is then undetermined until after the round trip of the transmission line, thus slowing down the response of the system.

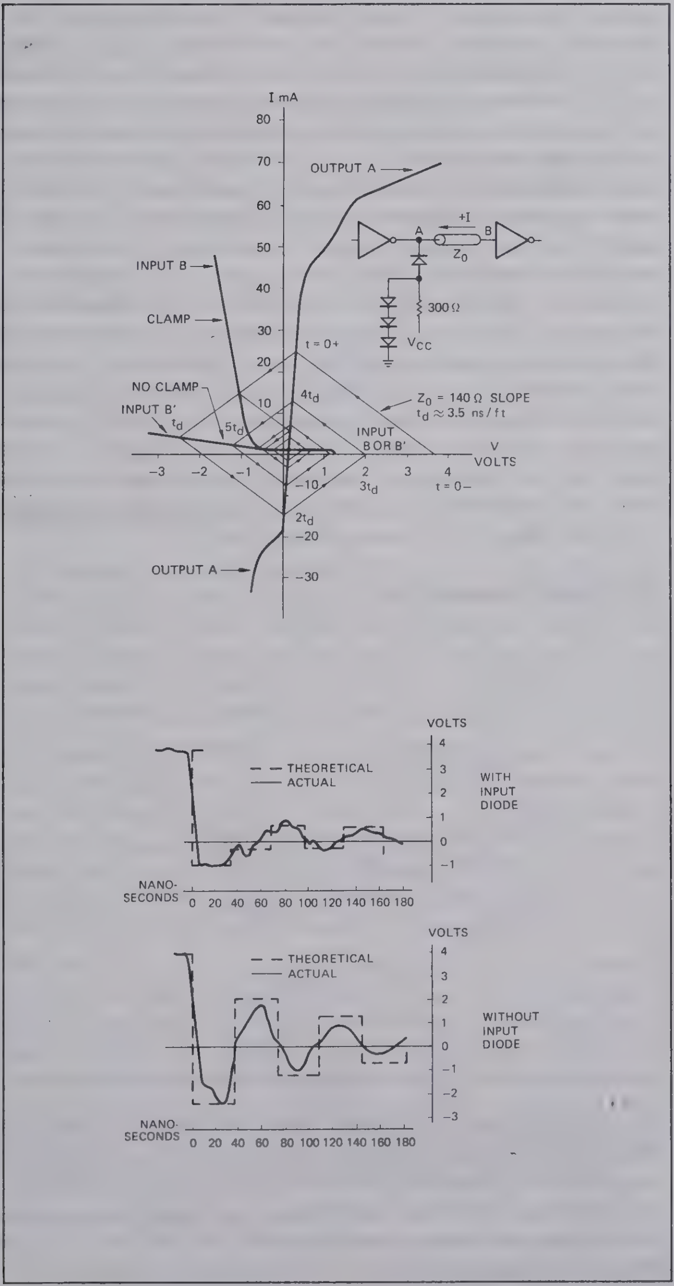
TTL Driving Transmission Line



If V_{OUT} is increased to > 2 V by either increasing Z_O or decreasing R_O, additional delay does not occur. R_O is a characteristic of the driver output configuration, varying between the different TTL speed categories. Z_O can be changed by varying the thickness of the conductor and its distance from ground. The following table lists the lowest transmission line impedance that can be driven by different TTL devices to insure an initial voltage step of 2 V. Note that the worst case value, assuming a +30% tolerance on the current limiting resistor and a –10% tolerance on V_{CC}, is 80% higher than the value for nominal conditions.

A graphical method provides excellent insight into the effects of high speed digital circuits driving interconnections acting as transmission lines. The method is basically to draw a load line for each input and output situation. Each load line starts at the previous quiescent point, determined where the previous load line cuts the appropriate characteristic. The magnitude of the slope of the load lines is identical and equal to the characteristic impedance of the line, but alternate load lines have opposite signs representing the change in direction of current flow. The points where the load lines cut the input and output characteristics represent the voltage and current value at the input or output, respectively, for that reflection. The method is shown below with and without the input diode, and illustrates how the input diode on TTL elements assists in eliminating spurious switching due to reflection.

Ringing Caused by Reflections



CROSSTALK

Crosstalk, the coupling of energy from one circuit to another via real and parasitic capacitance and inductance, causes increased problems in digital systems as the rise and fall times of the circuit decrease. The subject is extremely complicated, and no simple formula can give correct values in all cases for the amplitude of noise coupled from one circuit to another. In some circumstances where the input and output resistances of the circuits are high, a lumped equivalent circuit model can be drawn and reasonable calculations made. However, when the connections act as transmission lines, the situation is extremely complicated. TTL elements have a low output impedance in both High and Low logic states, and it is very difficult to couple enough energy into a short interconnection between devices to switch an adjacent circuit erroneously.

The effect of the fast rise and fall times of TTL devices is to increase the noise coupling between circuits. However, this greater noise coupling is more than offset by the low output impedance of TTL devices in both logic levels.

POWER SUPPLY FOR TTL SYSTEMS

A well regulated power supply should be used with ripple of ≤ 5% and regulation ≤ 5%. For large systems a transmission line low inductance power bussing system and an adequate amount of RF bypassing may be necessary.

DECOUPLING

Decoupling capacitors should be used on every PC card, at least one for every 5 to 10 standard TTL packages, one for every five 9H and 9S packages and one for every one-shot (monostable), line driver and line receiver package. They should be good quality RF capacitors of 0.01 μF to 0.1 μF with short leads. It is particularly important to place good RF capacitors near sequential (bistable) devices. In addition, a larger capacitor (preferably a tantalum capacitor) of 2 μF to 20 μF should be included on each card.

GROUNDS

A good ground system is essential for a PC card containing a large number of packages. The ground can either be a good ground bus, or better yet, a ground plane which, incorporated with the V_{CC} supply, forms a transmission line power system. Power transmission systems are commercially available which can be attached to a PC card to give an excellent power system without the cost of a multilayer PC card. Ground loops on or off PC cards are to be avoided unless they approximate a ground plane.

UNUSED INPUTS

Theoretically, an unconnected input assumes the High logic level, but practically speaking it is in an undefined logic state because it tends to act as an antenna for noise. Only a few 100 mV of noise causes the unconnected input to go to the logic Low state. On devices with memory (flip-flops, latches, registers, counters), it is particularly important to terminate unused inputs (MR, PE, PL, CP) properly since a noise spike on these inputs might change the contents of the memory. It is poor design practice to leave unused inputs floating. Instead, follow these important recommendations:

- Tie unused NAND, OR, AND inputs (multi-emitter inputs) to a used input of the same gate, provided the High level fan out of the driving circuit is not exceeded. Note that the Low level fan out is not increased by this connection because the inputs share a common base pull-up resistor.
- Tie unused NOR or OR inputs to ground or connect them to a used input of the same gate, provided the fan out of the driving circuit is not exceeded. Note that this connection increases both High and Low level fan in since each OR input has a separate single emitter input transistor.
- To provide a permanent logic Low on an input, tie it to ground.
- To provide a permanent logic High to an input:
  - — Tie the input V_{CC}, but note that V_{CC} must never, not even for a few ns, exceed 5.5 V. Inputs break down at some unspecified voltage about 5.5 V; the current then is not limited and the input and other circuitry on the chip may be permanently damaged.
  - — Tie the input to V_{CC} through a current limiting resistor of $\approx 1\ k\Omega$. One resistor can be used to define a High for up to 50 inputs.
  - — Tie the input to a separate power supply of 2.4 to 5.5 V if available.
  - — Tie the input to the output of a spare gate that is permanently High. Grounding one or more inputs of a NAND gate generates a permanently High output.

UNUSED GATES

It is recommended that the outputs of unused gates be forced High by tieing a NAND gate input or all NOR gate inputs to ground. This lowers the power dissipation and supplies a logic High at the gate output which can be used at unused inputs to other gates.

EXTENDERS

For maximum speed, TTL extender gates (9006) or discrete diodes should be placed as close as possible to the gate being extended. This practice minimizes capacitance at the sensitive extension points.

INCREASING FAN OUT

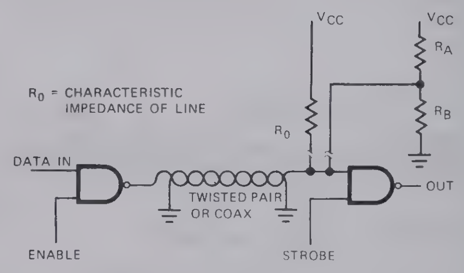
To increase fan out, inputs and outputs of gates on the same package may be paralleled. It is advisable to limit the gates being paralleled to those in a single package to avoid large transient supply currents due to different switching times of the gates. This is not detrimental to the devices, but could cause logic problems if the gates are being used as clock drivers.

LINE DRIVING AND RECEIVING

Open wire connections between TTL circuits should not be bundled, tied, or routed together. Instead, point-to-point wiring should be used, preferably above a ground plane which reduces coupling between conductors.

Single line wire interconnections should not exceed two feet; for wires longer than 15 inches, a ground plane is essential to provide adequate system performance. Over 2-foot twisted pairs or coaxial cable should be used. The characteristic impedance of an open wire over a ground plane is about 150Ω, while for twisted pairs of #26 wire the impedance is about 120Ω. For added protection against crosstalk, coaxial cables can be used but coaxial cables having very low characteristic impedances are difficult to drive. For best performance, coaxial cables with a characteristic impedance R_o of 100 Ω should be used. Resistive pull ups at the receiving end can be used to increase noise margins. If reflection effects are unacceptable, the line must be terminated in its characteristic impedance. One method is shown below where the output of the line is tied to V_{CC} through a resistor equivalent to the characteristic impedance of the line. Therefore R_o is fairly small, and the driving gate must sink the current through it in addition to the current from the inputs being driven. Terminating the line in a voltage divider with two resistors, each twice the line impedance, reduces the extra sink current by 50%. It is preferable to dedicate gates solely for line driving if the line length is in excess of five feet.

TTL Driving Twisted Pair



$$R_A = R_B = 2\ Z_o$$
$$R_o = Z_o$$

For additional noise immunity when driving long lines, the 9614 or 9615 dual differential line driver and line receiver may be used. These devices drive a twisted pair of wires differentially, permit easy termination of lines, and provide +15 V common mode noise rejection as described in Line Drivers and Receivers.

INTRODUCTION

The simplest data storage circuit is the cross-coupled gate (latch), perhaps with an Enable input, sometimes misleadingly called a clock. These single rank circuits are called transparent because the output can respond immediately to input data changes. Outputs may not be directly fed back to the inputs, since this creates a race condition (potential oscillation).

On the other hand, flip-flops and registers are nontransparent, dual-rank data storage circuits consisting of two sets of latches per bit controlled by a common clock. The master latch is affected by the input conditions *during* each clock pulse; the slave latch is controlled by the master *between* clock pulses. Only the slave outputs are brought out and they can change only immediately after the end of a clock pulse.

Because of the nontransparent, dual-rank nature of these circuits, the outputs can be fed back to the inputs without danger of race conditions. Multiphase clocking schemes are not required and system design is greatly simplified. To use these devices appropriately, the designer must, however, have a clear understanding of the timing parameters (output propagation delay, input timing requirements) and be aware of problems generated by clock delay variations (clock skew).

PROPAGATION DELAY

The outputs of any flip-flop or register change only as a result of a single clock edge — the Low-to-High clock transition in modern circuits. The delay between this clock edge and the output change is specified as t_{pd}. Usually the delays are different for the two possible directions of output change and specified as

t_{pd+} or t_{PLH}

the delay between the active clock edge and the output change from Low to High, and

t_{pd-} or t_{PHL}

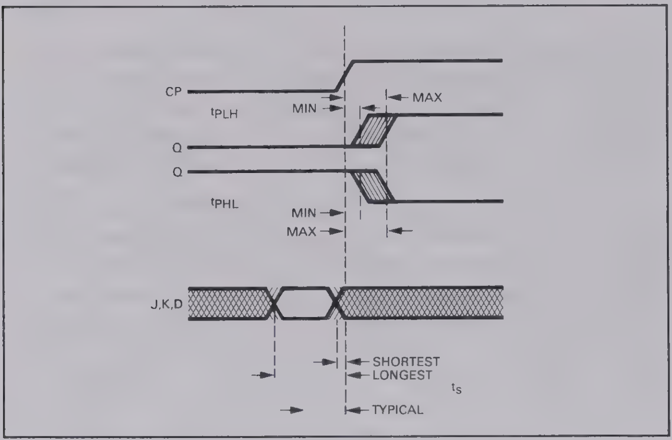
the delay between the active clock edge and the output change from High to Low.

The manufacturers usually give typical and maximum values; the minimum value must obviously be >0.

SET-UP TIME

Edge triggered flip-flops and registers are affected by synchronous input levels only at one moment, a short time before the active clock edge. This critical time is called the set-up time t_S. Depending on the input levels at that moment, the flip-flop or register is either set, reset, or left unchanged. The levels and level changes on synchronous inputs at any other time do not affect the flip-flop or register. Since all device parameters vary with temperature, supply voltage, and manufacturing tolerances, set-up time has a certain spread. It is important to make sure the synchronous input levels remain stable during this timing window between the longest t_S (representing a slow device) and the shortest t_S (representing a fast device).

Timing



Semiconductor manufacturers have, at various times, used different and even conflicting nomenclature for the two borders of this timing window. The longest set-up time is often called $t_{s(max)}$, but sometimes $t_{s(min)}$ since it describes the shortest time the inputs must remain stable before the next clock edge. The shortest set-up time is called $t_{s(min)}$, but also, less logically, $t_{s(max)}$. Fairchild uses the term t_r (release time) and other manufacturers call it $-t_{hold}$ (negative hold time). This name originated with early circuits (like the 7470) where the clock delay was longer than the data delay. These circuits required that the data inputs be held stable even after the active clock edge (positive hold time). All modern circuits have positive set-up times and therefore negative hold times.

ONE CATCHING MASTER/SLAVE FLIP FLOPS

As shown in the flip-flop selection guide, a number of older flip-flops (9000, 9001, 9020, 9022, 74H71, 7472, 7473, 7476, 7478, 74104, 74105) are not edge triggered, but are true master/slave flip-flops. They consist of two latches, a master latch that accepts input data *during* the clock pulse and a slave latch controlled by the master *between* clock pulses. Only the outputs from the slave are brought out. All these flip-flops have JK inputs and they toggle when both J and K are activated. This is achieved by ANDing the J input with the $\overline{Q}$ output to generate the Set input to the master, and ANDing the K input with the Q output to generate the Reset input to the master. Thus a one in the slave latch prevents the J input from setting the master, while a zero in the slave latch prevents the K input from resetting the master. During the clock pulse, the master thus searches for an input that causes it to change state. If it receives such an input signal, even a very short one, it accepts it, changes state and can then no longer be affected during this clock pulse. This characteristic is called 'one catching' (better, 'opposite state catching') and can result in improper operation if the synchronous inputs are unsettled during the clock pulse. For these master/slave flip-flops, the clock pulses should be as short as possible, accommodating all combinatorial delays and potential noise pulses between clock pulses. These considerations do not apply to edge triggered flip-flops (9024, 7470, 7474, 74H101, -102, -103, -106, -108, 74S112, -113, -114) and to the edge triggered registers and counters of the 9300 family.

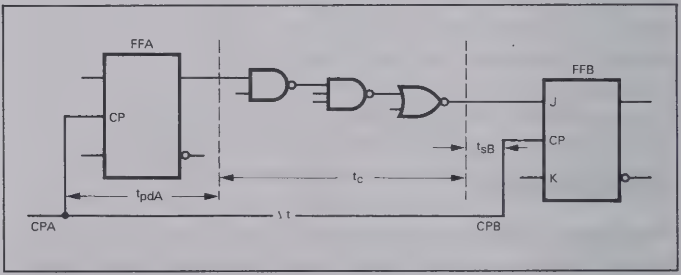
As stated before, these devices are only affected by the state of their synchronous inputs one set-up time before the active clock edge. They are, therefore, insensitive to the duty cycle of the clock. The clock to any flip-flop or register, master/slave or edge triggered, is a dc function. The devices are not directly sensitive to the clock rise and fall times, as are capacitance coupled flip-flops. However, if the rise or fall times are very long and the clock signal lingers around the threshold level longer than the flip-flop delay time, multiple clocking can result from the slightest noise on the clock line, or even from internal crosstalk on the chip. The clock signal should therefore generally have a transition time of less than 50 ns.

CLOCK SKEW

Clock skew is a problem inherent to any synchronous digital system using high speed TTL components. Independent of the system clock rate and affected only by the device speed, it often comes as an unpleasant surprise to the unsuspecting designer of low speed systems.

Ideally, all clock inputs in a synchronous system should be activated simultaneously; in a large system, this is obviously impossible. There are time differences, called clock skew, due to line propagation time and/or delay variations between different clock buffers.

Critical Timing



Consider the simple case of two flip-flops, FF A driving FF B through a combinatorial network with delay t_c. If the clock pulse arrives at FF B later than at FF A, this delay must be less than

$$\Delta t_{max} = t_{pdA} + t_c + t_{sB}$$

for the system to operate properly. If it arrives later, it affects FF B with the new, perhaps changed, contents of FF A. Note that for a worst-cased design, the critical values of t_{pd}, t_c, and t_s are the minimum (shortest, fastest) values often not specified in the data sheets.

There are two ways to avoid clock skew problems. One is to minimize clock skew by driving all clock inputs from one source, perhaps a parallel connection of several gates or buffers. (Only outputs from the same chip should be paralleled to minimize power supply glitches). The other method is to arrange clock delays carefully so the clock delays run in the opposite direction of the data delays. This insures that the last device in the data chain is clocked first, and all clock skew problems are eliminated. If this is impossible, e.g., in a recirculating register, a flip-flop with opposite clock polarity can be inserted in the data stream.

INTRODUCTION

All circuits in the Fairchild TTL family, in fact all TTL devices presently manufactured, are compatible. Any TTL output can drive a certain number of TTL inputs, as described in the loading rules. There are only subtle differences in the worst case noise immunity when low power, standard, and Schottky TTL circuits are intermixed. Open collector outputs, however, require a pull-up resistor to drive TTL inputs reliably.

While TTL is the dominating logic family, and many systems use TTL exclusively, there are cases where different semiconductor technologies are used in one system, either to improve the performance or to lower the cost, size, and power dissipation. The following explains how TTL circuits can interface with DTL, ECL (CML), CTL, and discrete transistors. Interfacing with MOS is not covered, since it depends too much on the particular MOS technology (high or low threshold, metal or silicon gate, p-channel or n-channel or CMOS).

INTERFACING TTL AND DTL

Both DTL and TTL are current sinking families, operating on a +5 V supply. They interface perfectly. When TTL drives DTL, one DTL input represents 1 UL in the Low state, much less than 1 UL in the High state. When DTL drives TTL, a 2 kΩ output has a drive capability of 8 UL, a 6 kΩ output has a drive capability of 4 UL.

INTERFACING TTL AND ECL

Mixing ECL and TTL logic families offers the design engineer a new level of freedom and opens the entire VHF frequency spectrum to the advantages of digital measurement, control and logic operation.

The chief advantages of emitter coupled logic are high speed, flexibility, design versatility and transmission line compatibility. But application and interfacing cost problems have traditionally discouraged the use of ECL in many areas, particularly in low cost, less sophisticated systems. Using 95K fully compensated ECL with new ECL/TTL interface devices and several new interfacing methods promises to extend the advantages of ECL to many low cost systems designs.

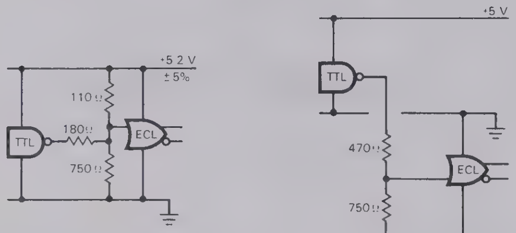
The most practical interfacing method for smaller systems involves using a common supply of +5 to +5.2 V. Care must be exercised with both logic families when using this technique to assure proper bypassing of the power supply to prevent any coupling of noise between circuit families. If only a few 95K ECL packages are designed into a predominantly TTL system, the safest method is to use a 0.01 µF miniature ceramic capacitor across each 95K device. This value capacitor has the highest Q, or bypassing efficiency. When larger systems are operated on a common supply, separate power busses to each logic family help prevent problems. Otherwise, good high frequency bypassing techniques are usually sufficient.

95K series ECL devices are fully compensated so that input thresholds and output levels are immune to broad variations in ambient temperature and supply voltage. This feature makes it easier to interface with TTL and to operate with the TTL power supply. 95K devices have high input impedance with input pull-down resistors (>20 kΩ) to the negative supply. In the TTL to ECL interface circuits on the following page, it is assumed that the ECL devices have high input impedance.

9500 series ECL elements are temperature compensated and have internal 2 kΩ pull-down resistors at each input and output. These resistors provide partial termination of interconnecting transmission lines, in many cases, eliminating the need for external terminations. For ECL inputs with 2 kΩ pull-down resistors, the 750 Ω resistors shown in the TTL to ECL circuits should be changed to 1.2 kΩ to provide the proper ECL input signal levels.

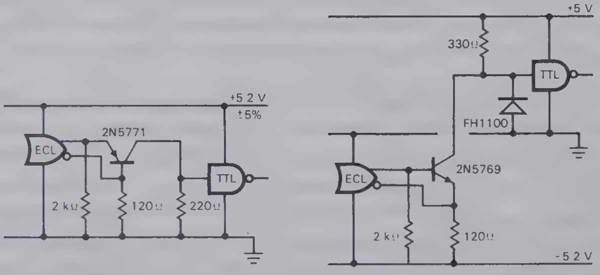
All circuits described operate with ±5% ECL and ±10% TTL supply variations, except those with ECL and TTL on a common supply. In those cases the supply can be ±10% with 95K ECL, ±5% with 9500 series ECL. All resistors are ¼ watt, ±5% composition type.

TTL to ECL Conversion


+5.2 V
±5%
110 Ω
180 Ω
750 Ω
TTL
ECL
+5 V
TTL
470 Ω
ECL
750 Ω
5.2 V

TTL to ECL conversion is easily accomplished with resistors, which simultaneously attenuate the TTL signal swing, shift the signal levels, and provide low impedance for damping and immunity to stray noise pick-up. The resistors should be located as near as possible to the ECL circuit for optimum effect. The circuits above assume an unloaded TTL gate as the standard TTL source. ECL input impedance is predominately capacitive (≈ 3 pF); the net RC time constant of this capacitance with the indicated resistors assures a net propagation delay governed primarily by the TTL signal.

ECL to TTL Conversion


+5.2 V
±5%
2N5771
ECL
TTL
2 kΩ
120 Ω
220 Ω
+5 V
330 Ω
TTL
FH1100
ECL
2N5769
2 kΩ
120 Ω
-5.2 V

Common Power Supply

Separate Power Supplies

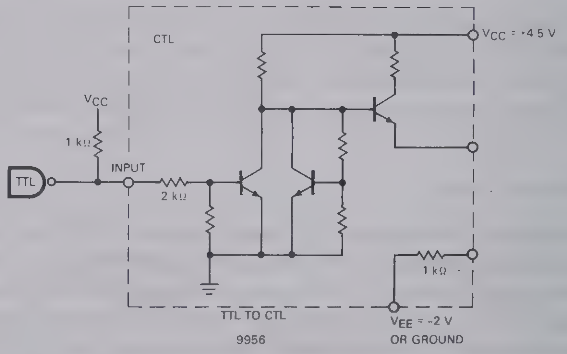
When interfacing between high voltage-swing TTL logic and low voltage-swing ECL logic, the more difficult conversion is from ECL to TTL. This requires a voltage amplifier to build up the 0.8 V logic swing to a minimum of 2.5 V. The circuits shown below may be used to interface from ECL to TTL.

The higher speed converters usually have the lowest fan out —only one or two TTL gates. This fan out can be increased simply by adding a TTL buffer gate to the output of the converter. Another option, if ultimate speed is required, is to use additional logic converters.

INTERFACING TTL AND CTL

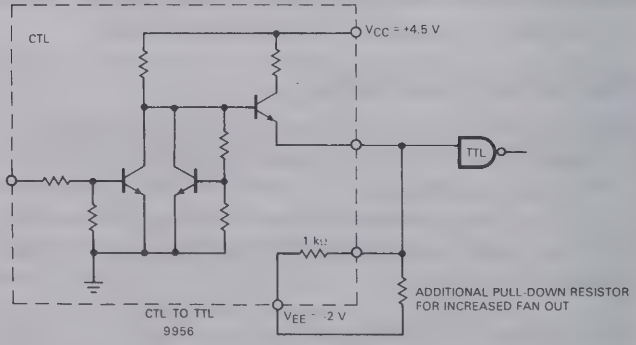
CTL (Complementary Transistor Logic) is a family of high speed digital circuits used mainly in computers. It uses AND gates and wired-OR outputs for logic flexibility, but logic levels are not restored in each gate. Level restoring buffers (9956) are therefore required, and all interfacing should be done with restored logic levels.

TTL to CTL Conversion


CTL
VCC = +4.5 V
VCC
1 kΩ
INPUT
TTL
2 kΩ
1 kΩ
TTL TO CTL
9956
VEE = -2 V
OR GROUND

The CTL input threshold is ≈ 1 V, similar to TTL, but 1 to 2 mA are required to pull the CTL input reliably over the threshold. A normal TTL output can drive a CTL input, but noise immunity is improved considerably by a 1 kΩ pull up resistor.

CTL to TTL Conversion


CTL
VCC = +4.5 V
TTL
1 kΩ
ADDITIONAL PULL-DOWN RESISTOR
FOR INCREASED FAN OUT
CTL TO TTL
9956
VEE = -2 V

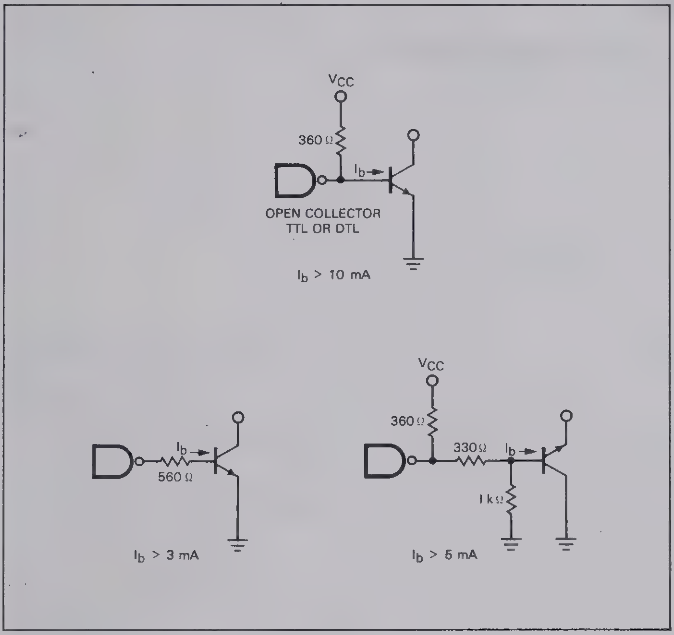
The CTL output emitter follower can source > 30 mA but cannot sink current. A resistive termination is therefore required. When the resistor is returned to ground, it may not exceed 250 Ω to guarantee a V_{OL} of < 400 mV at a fan out of 1 UL.

A better, less power consuming way for a fan out of 1 UL is to use the built-in pull down resistor (1 kΩ to –2 V). For increased fan out, this resistor can be reduced by a parallel external resistor to 180 Ω (8 UL).

TTL DRIVING TRANSISTORS

Although high voltage, high current ICs, such as the 9644, are available, it is sometimes necessary to control greater currents or voltages than integrated circuits are capable of handling. When this condition arises, a discrete transistor with sufficient capacity can be driven from a TTL output. Discrete transistors are also used to shift voltages from TTL levels to logic levels for which a standard interface driver is not available.

TTL Driving NPN Transistors



The three circuits above show how TTL can drive npn transistors. The first circuit is the most efficient but requires an open collector TTL or DTL output. The other two circuits limit the output current from the TTL totem pole output through a series resistor. The last circuit increases the High drive capability of the TTL output with an external pull up resistor and uses a base return resistor for faster turn off and better thermal stability.

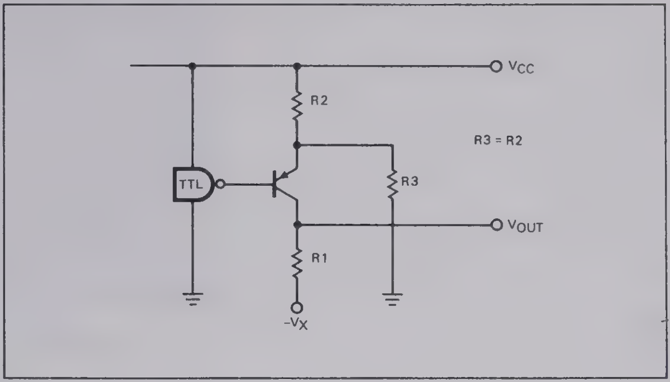
SHIFTING A TTL OUTPUT TO NEGATIVE LEVELS

This circuit uses a pnp transistor to shift the TTL output to a negative level. When the TTL output is High, the transistor is cut off and the output voltage is $-V_X$. When the TTL output is Low, the transistor conducts and the output voltage is

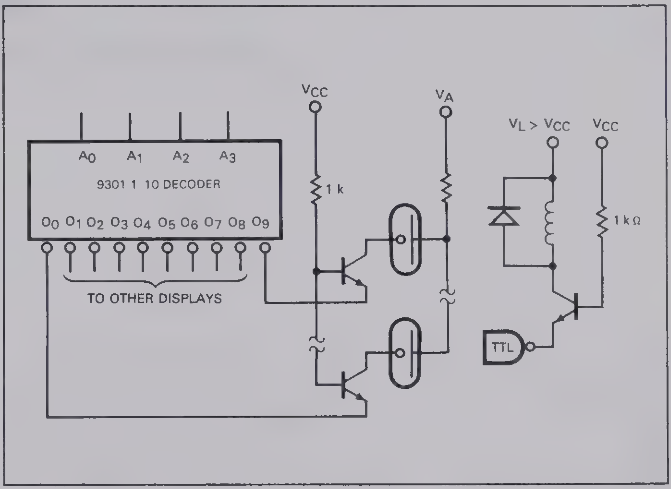
$$-V_X + \frac{R_1}{R_2}(V_{CC} - 2\text{ V})$$

if the transistor is not saturated, or slightly positive if the transistor is allowed to saturate.

PNP Transistor Shifting TTL Output

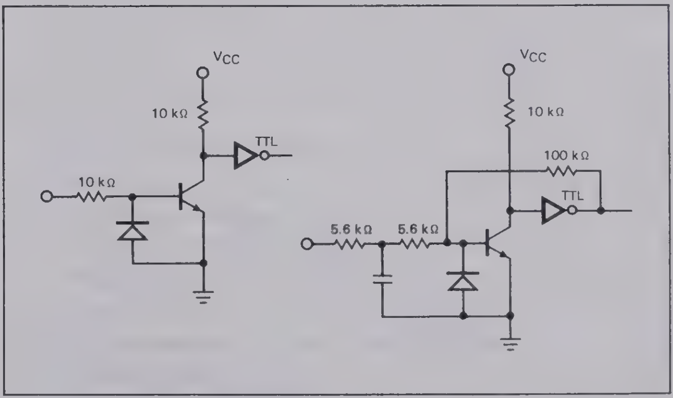


Non-Inverting High Voltage Drivers

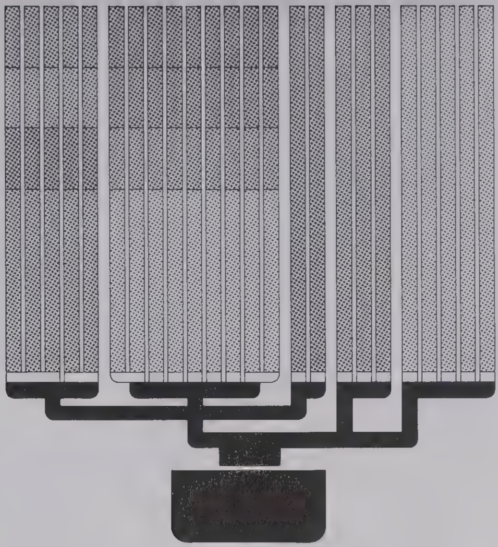


A TTL output can be used to drive high voltage, low current loads through the simple, non-inverting circuits shown above. This can be useful for driving gas discharge displays or small relays, where the TTL output can handle the current but not the voltage.

TRANSISTORS DRIVING TTL



It is sometimes difficult to drive the relatively low impedance and narrow voltage range of TTL inputs directly from external sources, particularly in a rough, electrically noisy environment. The circuits shown above can handle input signal swings in excess of ± 100 V without harming the circuits. The second circuit has an input RC filter that suppresses noise. Unambiguous TTL voltage levels are generated by the positive feedback (Schmitt trigger) connection.



FUNCTIONAL CROSS REFERENCE

AC Noise Margins, TTL . . . . 15-8 *f*

Accumulator
  For Arithmetic Logic Unit . . . . 5-28 *ff*, 8-27
  8-Word . . . . 8-22 *ff*
  16-Digit Decimal (BCD) 5-14
  16-Word 8-25
Active, Definition 5-5

Adder 5-4
  (See also Adders/Subtractors)
  Dual 5-8
  Serial Binary, For Many Variables 5-10
  1-Bit 1-19
  2-Bit and 1-Bit 5-12
  3-Bit 5-12
  4-Bit 5-12

Adder/Subtractor
  Carry Lookahead . . . . 5-21, 5-25 *ff*
  Complement Number Representation 5-20
  Decimal (BCD) 5-13
  Decimal or Mixed Code 5-15
  Serial Binary 5-9
  Serial Decimal (BCD) 5-11
  Sign Magnitude 5-20
  4-Bit . . . . 5-16 *ff*, 5-22 *ff*
Addition of Signed Numbers 5-7
Addressable Latch, 8-Bit . . . . 6-13 *ff*
Addressing Techniques, Memory 7-3
Advantages of 9300 Family XII
Analog to Digital Converter . . . . 4-11, 6-20 *f*
Arithmetic
  (See Adders/Subtractors, Multipliers/Dividers)

Arithmetic Logic Unit 5-4
  4-Bit . . . . 5-16 *ff*, 5-22 *ff*
  With Carry Lookahead . . . . 5-21, 5-25 *ff*
  With 3 Registers . . . . 5-28 *f*
  With 8 Registers . . . . 5-30 *f*, 8-27
Astable Oscillators 10-5, 13-14
BCD
  (See Decimal)
Blanking, Display 3-7
Bounce Eliminator
  (See Switch Bounce Eliminator and Switch Encoder)
Carry, Lookahead 5-6
  (See also Arithmetic Logic Unit, Carry Lookahead)
Carry, Ripple 5-5
Circuit Description, Gate . . . . 15-4 *f*
Clock
  Driver, Discrete, For MOS 6-23
  Generator 10-5, 13-14
  Generator, 4-Phase 2-7
  Generator, 8-Phase 6-23
  Monitor 13-14
  Skew 15-14
CMOS, Comparison to Other Logic Families IX
Code Converters . . . . 5-55 *ff*
  Angle-to-Decimal (BCD) 5-59
  Binary-to-Decimal (BCD), Serial 5-57
  Decimal (BCD)-to-Binary 5-55
  Decimal (BCD)-to-Binary, Serial 5-56
  Decimal (BCD)/Binary . . . . 5-58 *f*
  General Scheme 4-9
  Gray/Binary 5-60
Commutator 4-14
Commutator, Data 1-10

FUNCTIONAL CROSS REFERENCE

Comparator
  Four Word Sorter 1-7
  Identity 5-45
  Magnitude 5-5, 5-48 *ff*
  Magnitude, Serial 5-46 *f*
  Magnitude, Using Arithmetic Logic Unit 5-22, 5-27
Complement Generator, Nine Complement 5-61, 9-21
Contact Bounce Eliminator
  (See Switch Bounce Eliminator and Switch Encoder)
Control Sequencer 1-14
Converter
  (See Analog-to-Digital, Digital-to-Analog, Serial-to-Parallel, Parallel-to-Serial, or Code)
Correction, Error 5-51 *ff*
Cost of MSI Systems XI

Counter Chapter 9
  Survey 9-3 *ff*
  Dead End 9-26, 9-34
  Decimal 9-6 *ff*
  Decimal (BCD) 9-17 *ff*
  For Stopwatch, With Display 9-40
  Johnson 9-35, 10-8
  Modulo Variable 9-10
  Modulo 11, 13, or 15 9-11
  Modulo 12 9-6 *ff*
  Modulo 15, Shift 9-37
  Modulo 16 9-6 *ff*, 9-17 *ff*
  Modulo 16, Shift 8-21
  Any Modulo From 21 to 127 9-11 *ff*
  Modulo 32 and 64 9-22
  Modulo 525 and 625 9-15
  Moebius 9-35, 10-8
  Multistage 9-10, 9-20, 9-23 *ff*
  Programmable 9-17 *ff*, 9-21 *ff*, 9-28, 9-33
  Programmable, 2 n 2-22
  Programmable, Shift 9-37 *f*
  Programmable, Super High Speed 9-26
  Shift Register, Combination 9-27
  Shift, Synchronous 9-35 *ff*
  Synchronous 9-16 *ff*
  Universal 9-6 *ff*
  Up/Down 9-30 *ff*
  Up/Down Shift 7-13, 9-37
  Up/Down, for Asynchronous Inputs 9-34
  Up/Down, Incrementer 9-39
  Up/Down, Johnson 9-35
  Up/Down, Light Controlled 9-33
  Up/Down, Multistage 9-32
  Up/Down, Single Line Control 9-34
  With Multiplexed Display 8-20, 9-39
  With Switch Bounce Eliminator 9-14
  With 50% Duty Cycle 9-6 *ff*, 9-22
  4-Stage 9-6 *ff*
Crosstalk 15-11
CTL, Comparison to Other Logic Families IX
CTL, Interfacing With TTL 15-16
Current, Input/Output, TTL 15-7
Darlington Output, TTL 15-5

Data
  Bussing, Multiple Word 1-10
  Commutator 1-10
  Latching 6-11
  Routing 1-5, 1-12
  Transmission Chapter 14
DC Noise Margins, TTL 15-7 *ff*

FUNCTIONAL CROSS REFERENCE

Decimal
  Accumulator, 16 Digit 5-14
  Adder/Subtractor 5-13
  BCD-to-Binary Code Converter 5-55
  BCD/Binary Code Converter .... 5-58 *f*
  BCD/Binary Code Converter, Serial .... 5-56 *f*
  Counter ....9-6 *ff*
  Counter, Shift......................................9-36
  Counter/Shift Register 9-27
  Encoder 4-7
  Multiplexer 1-15
  Multiplexer, Dual...................................1-6

Decoder ....Chapter 2
  Dual 1-of-4 2-4
  1-of-8 ....2-8, 6-13
  1-of-10....................................2-8, 2-12
  1-of-10, Driver 2-14
  1-of-16...................................2-15, 6-19
  1-of-32...2-16
  1-of-64 ....2-6, 2-9
  1-of-80...2-9
  1-of-128 2-17
  1-of-1024 2-17
  7-Segment Display ....3-4 *ff*
  Any 4-Bit Code 2-20
Decoding, Address ....2-5, 2-9, 2-16 *ff*
Decoupling, Power Supply 15-11
Delay, Programmable Using Monostable 13-13
Delayed Pulse Generator 13-20

Demultiplexer
  (See also Decoders)
  Clock, 16-Channel 2-21
  Data 6-13, 6-19
  Data, With Storage 6-15
  Data, Dual 4-Channel 2-6
  Data, 8-Channel 2-10
  Data, 16-Channel 2-21
  Data, 32-Channel 2-10
Design Considerations, Device XI
Design Rules for Systems..............................X
Detection, Error ....5-51 *ff*
Detector
  Double Edge 13-20
  Edge 8-10, 10-4, 10-8
  1-of-8 (Single Zero) 4-14
Differentiator, Digital
  (See Edge Detectors)
Digital Noise Suppression...........................6-24

Digital-to-Analog Converter 9-29
  Binary..4-12
  Decimal (BCD) 4-13
Discrete Devices, Driving with TTL 15-17
Discriminator, Frequency 13-13

Display System ....Chapter 3
  Cold Cathode 3-18
  Gas Discharge 3-18
  Light Emitting Diode ....3-20 *f*
  Multiplexed ....3-8 *f*
  Multiplexed, Fluorescent ....3-16 *f*
  Multiplexed, for Stopwatch 9-40
  Multiplexed, Gas Discharge 3-19
  Multiplexed, Incandescent 3-13
  Multiplexed, Light Emitting Diode ....3-22 *ff*
  Multiplexed, With Code Converter 5-59
  Multiplexed, With Counter 8-20, 9-39

FUNCTIONAL CROSS REFERENCE

Display System, Cont.
  Multiplexed, 8-Digit 3-10 *ff*
  7-Segment Decoder 3-4 *ff*
  7-Segment Flourescent 3-14 *f*
Distortion, Line 14-7 *ff*
Divider, Binary 5-38 *f*
Divider, Frequency
  (See Counters)
Division Using Counters 5-32
Double Pulse Detector 13-13
Drive Capability of TTL, Transmission Line 15-9 *ff*
Driver, Discrete, TTL to Relay, Motor, Etc. 6-19
Driver, 7-Segment Display 3-4 *ff*
DTL, Comparison to Other Logic Families IX
DTL, Interfacing With TTL 15-15
ECL, Comparison to Other Logic Families IX
ECL, Interfacing With TTL 15-15 *f*

Edge Detector 8-10, 10-4, 10-8
  Double Edge 13-20
  Switch Synchronizer 5-58
Edge Triggered, Definition 10-6
EIA RS 232-C Interface 14-14
Electrical Properties of TTL Circuits Chapter 15

Encoder Chapter 4
  Priority Linear 4-8
  Priority, 8-Input 4-4
  Priority, 10-Input 4-7
  Priority, 16-Input 4-5
  Priority, 20-Input 4-7
  Priority, 64-Input 4-6
  Thumbwheel Switch 2-13
Error Correction 5-51 *ff*
Error Detection 1-19, 5-51 *ff*

Excess-3 Code Decoding 2-20
Excess-3 Gray Code Conversion 5-60
Eye Pattern, Binary 14-7 *ff*
False, Definition 5-5
Fan Out, Increasing TTL Drive 15-12

Flip-Flop
  Characteristics 15-13 *f*
  Multiple Slave 2-22
  Quad D 8-8, 9-26
  Survey 10-6 *f*

Frequency Discrimination 13-13
Frequency Divider
  (See Counters)
Function Generator 1-18, 2-7, 2-11
Gates, Increasing Drive 15-12
Generator, Clock 10-5, 13-14
Gray Code Decoding 2-20
Gray Code/Binary Code Converter 5-60
Hamming Code 5-51
Hamming Code Generator/Checker 5-52
High Speed TTL, Schottky Chapter 12
High Voltages, Interfacing With TTL 15-17
High, Definition 5-5
Hold Time, Definition 15-14
Holding Register 6-9, 6-12
Identity Comparator 5-45
Impedance, Input/Output, TTL 15-6
Incrementers
  (See Counters)
Input Characteristics, TTL 15-6
Inputs, Unused TTL 15-11 *f*

FUNCTIONAL CROSS REFERENCE

Interface .... Chapter 14, 15-15 *ff*
  Discrete, TTL to Relay, Motor, Etc. 6-19
  From TTL 10-3
  MIL STD 188C (Low Level) 14-14
Interrupt, Priority 4-10
JK Inputs, Truth Table for 10-6
Johnson Counter 9-35, 10-8
Keyboard Encoder 1-17, 2-20, 4-8
Keyed Oscillator 13-20
Latch .... Chapter 6
  Addressable, 8-Bit .... 6-13 *ff*
  Addressable, 16/32/64-Bits 6-16
  Array .... 6-7, 6-17 *f*
  Master/Slave
    (See Registers)
  Multiplexing Outputs 6-9
  R/S .... 6-10 *f*
  Selection 6-6, 6-12
  Survey 6-4
  Dual 2-Bit 6-25
  4-Bit .... 6-10 *f*
  Dual 4-Bit 6-5
Led Display
  (See Display)
Levels, Worst Case, TTL 15-6
Line Distortion .... 14-7 *ff*
Line Drivers .... Chapter 14
Line Driver, Selection of 14-2, 14-11
Line Receivers .... Chapter 14
Line Receiver, Selection of 14-2, 14-11
Linear Feedback Shift Counter 9-36
Loading, Effect of Collector And/Or 10-3
Logic Implementation, Random 1-18
Lookahead, Carry 5-6
Low Power TTL .... Chapter 11
Low Power, Mixing With Standard Power TTL 11-3
Low, Definition 5-5
Magnitude Comparator
  (See Comparator, Magnitude)
Malfunction Detector 13-14
Master/Slave, Definition 10-6

Memory .... Chapter 7
  (See also Latches, Registers, and Shift Registers)
  Addressing Techniques 7-3
  Array 7-11
  Buffer 7-9, 7-12
  First In/First Out 8-28
  For Display, Relays .... 6-17 *f*
  For Multiplexed Display 8-20
  Last In/First Out 7-13, 8-28
  Organization 7-3
  Push Down Stack 7-13, 8-28
  Random Access, 256-Bit 7-10
  Random Access, 1024-Bit 7-14
  Read Only .... 7-15 *ff*
  Read Only, Array 7-18
  Read Only, Odd Configurations .... 7-19 *f*
  Read Only, 32-Word x 8-Bit 7-16
  Read Only, 256-Word x 4-Bit 7-17
  Scratchpad 7-7, 7-9, 7-12, 8-27
  Serial 8-17
  Serial Random Access 1-13
  Serial, 4096-Bit 6-23
  Timing Chart 7-8
  Timing Considerations .... 7-4 *ff*

FUNCTIONAL CROSS REFERENCE

Memory, Cont.
  Use of Bipolar for Main Memory 7-9
  Use of Random Access in Place of Read Only 7-9
MIL STD 188C Interface 14-14
Minterm Generator 2-7, 2-11
Missing Pulse Detector 13-14
Mixed Logic Functions 1-18
Modulation, Pulse Duration 13-13
Moebius Counter 9-35, 10-8

Monostable Multivibrators Chapter 13
  Avoiding Use of X
  Characteristics 13-6
  Non-Retriggerable 13-21 *ff*
  Non-Retriggerable Operation 13-4
  Programmable Delay 13-13
  Resettable 13-16 *ff*
  Retriggerable 13-9 *ff*
  Retriggerable, Dual 13-16 *ff*
  Retriggerable, Resettable 13-16 *ff*
  Special Considerations 13-8
  Timing Equations 13-4
  Timing Nomograph 13-5
  Triggering Condition 13-3
  Using Electrolytic Timing Capacitor 13-7
  Voltage Controlled 13-15
MOS, Comparison to Other Logic Families IX

MSI
  Advantages of X
  Advantages of 9300 Family XII
  Device Design Considerations XI
  Impact on Logic Design IX
Multi-Word Comparator/Sorter 1-7

Multiplex Data Transmission 14-4 *f*
Multiplexed Display
  (See Display, Multiplexed)

Multiplexer Chapter 1
  Quad 2-Input 1-4
  Dual Independent 2-Input 1-10
  Dual 4-Input 1-8 *f*
  8-Input 1-11
  10-Input 1-15
  Dual 10-Input 1-6
  16-Input 1-16
  Dual 16-Input 1-10
  32-Input 1-12, 1-15
  64-Input 1-12

Multiplier
  Binary 5-34 *f*
  Combinatorial 5-5, 5-40 *ff*
  Combinatorial Array 5-41 *f*, 5-44
  Decimal (BCD) 5-36 *f*
  Serial Binary 5-33
  Two Complement 5-43 *f*
  Using Counters 5-32
  4 by 2-Bit 5-40, 5-43
Names of Inputs/Outputs XII,XIII
Negative Levels, From TTL Outputs 15-17
Negative Number Representation 5-3, 5-6
Nine Complement Generator 5-61, 9-21
Noise Margins, TTL 15-7 *ff*
Noise Suppression, Digital 6-24
Nomenclature for Inputs/Outputs XII, XIII
Number Representation 5-3, 5-6
One, Definition 5-5

FUNCTIONAL CROSS REFERENCE

- One Catching 6-11
- One Catching, Definition 10-6, 15-14
- One Complement Number Representation 5-6
- One Shots Chapter 13
  - (See Monostable)
- Open Collector Output, TTL 15-5
- Open TTL Inputs 15-11 *f*
- Operators Chapter 5

Oscillator

  - Clock 10-5, 13-14
  - Keyed, Synchronized 13-20
  - Voltage Controlled 10-5
- Output Characteristics, TTL 15-6
- Output Configurations, TTL 15-5
- Output Swing, How to Increase 10-3
- Overflow in Addition and Subtraction 5-7
- Package, Constraint on MSI Devices XI
- Parallel-to-Serial Converter 8-11
- Parallel-to-Serial Converter, 8-Bit 8-13
- Paralleling Gates 15-12
- Parity Bit 5-51

Parity Generator/Checker 5-5

  - Serial 5-51
  - 12-Bit 5-53 *f*
- Pattern Detector 1-19
- Pattern Generator 1-17
- PC Board Interconnections 15-9 *ff*
- Power Dissipation, TTL 15-6
- Power Supply Decoupling 15-11
- Power, Constraint on MSI Devices XI
- Priority Interrupt 4-10
- Priority Network
  - (See Encoder)
- Programmable Counter
  - (See Counter, Programmable)
- Programmable Delay Using Monostable 13-13
- Propagation Delay, Definition 15-13
- Pseudo-Random Sequence Generator 8-21
- Pulse Detector, Double Pulse 13-13
- Pulse Detector, Missing Pulse 13-14
- Pulse Duration Modulation Generation/Detection 13-13
- Pulse Shaper, Schmitt Trigger 13-14
- Push Down Memory 7-13, 8-28
- RAM
  - (See Memory)
- RAM, Definition 7-3
- Random Access Memory
  - (See Memory)
- Rate Multiplier, Binary 4-12
- Rate Multiplier, Decimal (BCD) 4-13, 9-16
- Read Only Memory
  - (See Memory)
- Readout
  - (See Display)

Register Chapter 8

  - (See also Shift Register)
  - Characteristics 15-13 *f*
  - Definition 8-3
  - For Arithmetic Logic Unit, Single Address 5-28 *f*
  - For Arithmetic Logic Unit, Three Address 5-30 *f*, 8-27
  - Master/Slave 2-22, 6-8
  - Multiple Port, 8-Bit 8-22 *f*
  - Multiple Port, 16-Bit 8-25

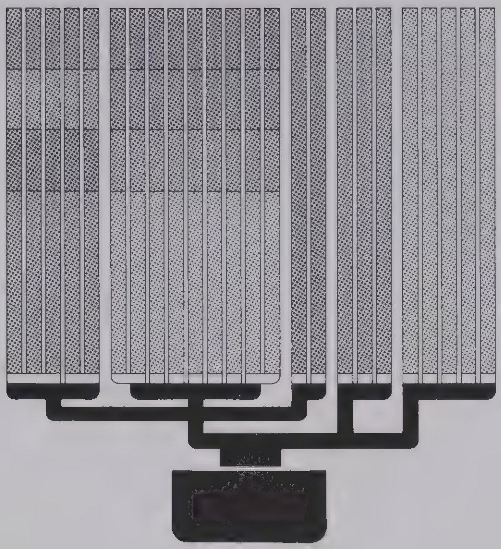
Register, Cont.
  Right Shift/Left Shift/Parallel Load 1-6
  Selection 6-6, 6-12
  Single Rank
    (See Latches)
  Survey .... 8-3 *ff*

Release Time, Definition 15-14
Retriggerable Monostable .... 13-9 *ff*
Retriggerable Monostable, Resettable, Dual .... 13-16 *ff*
Ringing, TTL 15-9
Ripple Carry 5-5
ROM
  (See Memory)
ROM, Definition 7-3
RTL, Comparison to Other Logic Families IX
Scanner, Thumbwheel Switch 2-13
Scanner, 15-Key 1-17
Scanner, 16-Key 2-20
Schmitt Trigger 13-14
Schottky TTL .... Chapter 12
Scratchpad Memory 8-27
Selection, Data Source 1-5
Sequence Generator 1-17
Sequence Generator, Pseudo-Random 8-21
Sequence Generator, TV Sync 9-15
Sequencer
  (See Multiplexers)
Sequencer 9-16
Sequencer, Control 1-14
Sequencer, Data 1-10
Serial
  (See Specific Function such as Adder/Subtractor, Serial)
Serial Decimal Accumulator, 16-Digit 5-14
Serial Memory
  (See Shift Registers)
Serial-to-Parallel Converter 6-22, 8-9
Serial-to-Parallel Converter, 8-Bit 8-12
Set-Up Time, Definition 15-13

Shift Register
  (See also Registers)
  Counter, Combination 9-27
  With Enable .... 8-10, 8-14, 8-16, 8-29 *f*
  Left/Right 8-8
  Memory 8-17
  Memory, 4096-Bit 6-23
  With Recirculate .... 8-10, 8-15 *f*
  Right/Left/Parallel Load 1-6
  Use of Memory as 5-14
  Variable Length .... 8-18 *f*
  Dual 2-Bit 8-8
  4-Bit .... 8-23 *ff*
  4-Bit Universal .... 8-6 *f*, 8-29 *f*
  5-Bit 8-35
  8-Bit .... 8-12 *f*, 8-31
  Dual 8-Bit .... 8-14 *f*
Sign Magnitude Number Representation 5-6
Simplex Data Transmission 14-4
Skew, Clock 15-14
Sorter, Four Word Comparator 1-7
SSI .... Chapter 10
Stepper, Control 1-14
Stopwatch 9-40
Subtraction of Signed Numbers 5-7
Subtractor
  (See Adder/Subtractor)

FUNCTIONAL CROSS REFERENCE

- Subtractor, 1-Bit 1-19
- Supply Voltage, TTL 15-6
- **Switch Bounce Eliminator** 6-11, 6-24, 10-8, 13-15
  - Synchronizer 5-58
  - With Counter 9-14
- Switch Encoder 1-17, 2-13, 2-20, 4-8
- Synchronized Oscillator 13-20
- Synchronizer 8-10
- Synchronizer, Switch 10-8
- Systems Considerations, TTL 15-9 *ff*
- Systems Design, Impact of MSI on IX
- Systems, Advantages of Using MSI in X
- Systems, Design Rules X
- Temperature Range, TTL 15-6
- Termination, Transmission Line 14-5 *ff*
- Terminology on Inputs/Outputs XII, XIII
- Transfer Function, TTL 15-7 *ff*
- Transistors, Driving from DTL 10-3
- Transistors, Driving from TTL 15-17
- **Transmission Line**
  - Drive Capabilities of TTL 15-9 *ff*
  - Interface Chapter 14
  - Driving Chapter 14
  - Driving with TTL 15-12
  - Termination 14-5 *ff*
- True, Definition 5-5
- **TTL**
  - Characteristics Chapter 15
  - Comparison to Other Logic Families IX
  - Driving Transistors with 15-17
  - Interfacing with CTL 15-16
  - Interfacing with DTL 15-15
  - Interfacing with ECL 15-15 *f*
  - Output Configurations 15-5
- TV Sync Generator 9-15
- Twisted Pairs of Wire, Driving Chapter 14
- Twisted Pairs of Wire, Driving with TTL 15-12
- Two Complement Number Representation 5-6
- Unused TTL Inputs 15-11 *f*
- Voltage Controlled Monostable 13-15
- Voltage Controlled Oscillator 10-5
- Voltage, Supply, TTL 15-6
- Voltage, Worst Case, TTL 15-6
- Wired-OR, Effect on Loading 10-3
- Worst Case Levels, TTL 15-6
- Zero Catching 6-11
- Zero Suppression, Display 3-7
- Zero, Definition 5-5



TTL/MSI PRODUCT INDEX

DEVICE	DESCRIPTION	PAGE
9300	4-Bit Universal Shift Register	8-6
9301	1-of-10 Decoder	2-8
9302	1-of-10 Decoder (Open Collector)	2-12
9304	Dual Full Adder	5-8
9305	Variable Modulo Counter	9-6
9307	7-Segment Decoder	3-4
9308	Dual 4-Bit Latch	6-5
9309	Dual 4-Input Multiplexer	1-8
9310	Decade Counter	9-17
9311	1-of-16 Decoder	2-15
9312	8-Input Multiplexer	1-11
9313	8-Input Multiplexer (Open Collector)	1-11
9314	4-Bit Latch	6-10
9315	1-of-10 Decoder/Driver	3-18
9316	4-Bit Binary Counter	9-17
9317	7-Segment Decoder/Driver	3-4
9318	8-Input Priority Encoder	4-4
9321	Dual 1-of-4 Decoder	2-4
9322	Quad 2-Input Multiplexer	1-4
9324	5-Bit Comparator	5-48
9328	Dual 8-Bit Shift Register	8-14
9334	8-Bit Addressable Latch	6-13
9338	8-Bit Multiport Register	8-22
9340	4-Bit ALU	5-16
9341,74181	4-Bit ALU	5-22
9342,74182	Carry Lookahead Unit	5-25
93S43	4-Bit by 2-Bit Multiplier	5-43
9344	4-Bit by 2-Bit Multiplier	5-40
9345	1-of-10 Decoder	2-14
9348	12-Input Parity Checker/Generator	5-53
9350	Decade Counter	9-4
9352,7442	1-of-10 Decoder	2-8
9353,7443	Excess 3, 1-of-10 Decoder	2-14
9354,7444	Excess 3 Gray, 1-of-10 Decoder	2-14
9356	Binary Counter	9-4

TTL/MSI PRODUCT INDEX

DEVICE	DESCRIPTION	PAGE
9357,7446/47	7-Segment Decoder/Driver	3-5
9358,7448	7-Segment Decoder	3-5
9359,7449	7-Segment Decoder	3-5
9360,74192	Up/Down Decade Counter	9-30
9366,74193	Up/Down Binary Counter	9-30
9368	7-Segment Decoder/Driver/Latch	3-5
9369	7-Segment Decoder/Driver	3-5
9370	7-Segment Decoder/Driver/Latch	3-5
93H72	4-Bit Shift Register	8-29
9375,7475	4-Bit Latch	6-25
9377,7477	4-Bit Latch	6-25
9383,7483	4-Bit Full Adder	5-12
9390,7490	Decade Counter	9-3
9391,7491	8-Bit Shift Register	8-31
9392,7492	Divide by 12 Counter	9-3
9393,7493	Binary Counter	9-3
9394,7494	4-Bit Shift Register	8-32
9395,7495	4-Bit Shift Register	8-33
9396,7496	5-Bit Shift Register	8-35
93141,74141	1-of-10 Decoder/Driver	3-18
93150,74150	16-Input Multiplexer	1-16
93153,74153	Dual 4-Input Multiplexer	1-9
93164,74164	8-Bit Serial to Parallel Converter	8-12
93165,74165	8-Bit Parallel to Serial Converter	8-13
93406	1024-Bit Read Only Memory	7-17
93410	256-Bit Read/Write Memory	7-10
93415	1024-Bit Read/Write Memory	7-14
93434,7488A	256-Bit Read Only Memory	7-16
9600	Monostable Multivibrator	13-16
9601	Monostable Multivibrator	13-9
9602	Dual Monostable Multivibrator	13-17
9603,74121	Monostable Multivibrator	13-21
74122	Monostable Multivibrator	13-24
74123	Dual Monostable Multivibrator	13-25

Equivalent Low Power and Schottky circuits are included with the basic product listing.

NOTES

FAIRCHILD DOMESTIC FIELD SALES OFFICES

HUNTSVILLE, ALABAMA
3322 So. Memorial Parkway 35801
Suite 92
Tel: 205-883-7020 TWX: 810-726-2214

PHOENIX, ARIZONA
4414 N. 19th Avenue 85015
Suite G
Tel: 602-264-4948 TWX: 910-951-1544

SAN FRANCISCO, CALIFORNIA
Village Corner, Suite L
El Camino and San Antonio Rds.
Los Altos, California 94022
Tel: 415-941-3150 TWX: 910-370-7952

LOS ANGELES, CALIFORNIA
6922 Hollywood Blvd. 90028
Suite 818
Tel: 213-466-8393 TWX: 910-321-3009

SANTA ANA, CALIFORNIA
2100 East 4th Street 92705
Suite 207
Tel: 714-558-1881 TWX: 910-595-1109

SAN DIEGO, CALIFORNIA
7380 Clairmont Mesa Blvd. 92111
Suite 204
Tel: 714-279-6021

DENVER, COLORADO
5115 South Valley Highway
Greenwood Gardens, Suite 240
Englewood, Colo. 80110
Tel: 303-770-2884 TWX: 910-935-0708

STAMFORD, CONNECTICUT
2777 Summers Street 06905
Tel: 203-348-7701 TWX: 710-474-1763

FT. LAUDERDALE, FLORIDA
3440 N.E. 12th Avenue 33308, Room 2
Tel: 305-566-7414 TWX: 510-955-9840

TAMPA, FLORIDA
12945 Seminole Blvd.
Florida Twin Towers Bldg. 2, Room 6
Largo, Florida 33540
Tel: 813-585-3892

ORLANDO, FLORIDA
211 N. Maitland Avenue
Suite 307
Altamonte Springs, Florida 32701
Tel: 305-834-7000 TWX: 810-850-0152

CHICAGO, ILLINOIS
9950 W. Lawrence Avenue
Room 311
Schiller Park, Ill. 60176
Tel: 312-671-4660 TWX: 910-227-0051

INDIANAPOLIS, INDIANA
4002 Meadows Drive 46205
Suite 216
Tel: 317-542-0641 TWX: 810-341-3432

FORT WAYNE, INDIANA
2118 Inwood Drive 46805
Suite 111
Tel: 219-483-6453 TWX: 810-332-1507

BLADENSBURG, MARYLAND
5801 Annapolis Road 20710
Suite 500
Tel: 301-779-0954 TWX: 710-826-9654

BOSTON, MASSACHUSETTS
167 Worcester Street
Wellesley Hills, Mass. 02181
Tel: 617-237-3400 TWX: 710-348-0424

DETROIT, MICHIGAN
Westland Office Plaza
33 300 Warren Avenue Suite 101
Westland, Mich. 48185
Tel: 313-425-3250 TWX: 810-242-2973

MINNEAPOLIS, MINNESOTA
4640 W. 77th Street 55435
Suite 347
Tel: 612-920-1030 TWX: 910-576-2944

WAYNE, NEW JERSEY
1200 Preakness Ave. 07470
Tel: 201-696-7070

ALBUQUERQUE, NEW MEXICO
First National Bank, Bldg. E, Suite 1507
5301 Central N.E. 87108
Tel: 505-265-5601 TWX: 910-989-1186

BINGHAMTON, NEW YORK
3215 E. Main St. Suite 7
Endwell, N.Y. 13760
Tel: 607-754-1094

MELVILLE, NEW YORK
275 Broadhollow Road 11746
Tel: 516-293-2900 TWX: 510-224-6480

ROCHESTER, NEW YORK
600 Kreag Rd.
Pittsford, N.Y. 14534
Tel: 716-385-1130

WAPPINGERS FALLS, NEW YORK
120 E. Main Street (Rt. 9) 12590
Tel: 914-297-4351 TWX: 510-248-7686

SYRACUSE, NEW YORK
333 E. Onondaga Street 13202
Tel: 315-472-3391 TWX: 710-541-0499

CLEVELAND, OHIO
6151 Wilson Mills Rd.
Suite 102
Highland Heights, Ohio 44143
Tel: 216-461-8288

DAYTON, OHIO
4812 Frederick Road 45414
Suite 101
Tel: 513-278-8278 TWX: 810-459-1803

TULSA, OKLAHOMA
5321 S. Sheridan Road 74145
Suite 15
Tel: 918-663-7131

PHILADELPHIA, PENNSYLVANIA
Fort Washington Industrial Park
1000 Center
Fort Washington, Pennsylvania 19034
Tel: 215-886-6623 TWX: 510-665-1654

DALLAS, TEXAS
725 So. Central Expressway
Suite B-6
Richardson, Texas 75080
Tel: 214-234-3391 TWX: 910-867-4757

HOUSTON, TEXAS
6440 Hillcroft 77036
Suite 507
Tel: 713-772-0200

SEATTLE, WASHINGTON
700 108th Avenue Northeast
Suite 211
Bellevue, Washington 98004
Tel: 206-454-4946 TWX: 910-443-2318

MILWAUKEE, WISCONSIN
1810 S. Calhoun
New Berlin, Wisconsin 53151
Tel: 414-782-8270

FAIRCHILD INTERNATIONAL FIELD SALES OFFICES

AUSTRALIA
Fairchild Australia Pty. Ltd.
420 Mt. Dandenong Road
P.O. Box 151
Croydon, Victoria
Tel: 723-4131 Telex: 79030846

BRAZIL
Fairchild Electronica Ltd.
Rua Da Consolacao 3542
Caixa Postal 30.407
São Paulo S.P., Brazil
Tel: 33-5891
33-2765
TWX: NBR 021-261

CANADA
Toronto Regional Office
FSC
17 Canso Road, Unit 6
Rexdale, Ontario M9W-4M1
Tel: 416-248-0285 TWX: 610-492-2700

Fairchild Semiconductor
1385 Mazurette Suite 3
Montreal 355, Quebec

FRANCE
Fairchild Semiconducteurs, S.A.
11, Rue Sainte Felicite
75 Paris 15,
Tel: 824-8494 Telex: 842-20614

HONG KONG
Fairchild Semiconductor (HK) Ltd.
135 Hoi Bun Road
P.O. Box 9575
Kwun Tong, Kowloon
Tel: 89 0271 Telex: 780-3531

ITALY
Fairchild Semiconducttori, S.p.A.
Viale Cortina d'Ampezzo 152
00135 Roma
Tel: 00396-3274006

Fairchild Semiconducttori, S.p.A.
Via Palmanova 67 II B
20132 Milano
Tel: 2899248 Telex: 043 34338

GERMANY
Fairchild Halbleiter GmbH
European Headquarters
6202 Wiesbaden Biebrich
Postfach 4559
Hagenauer Strasse 38
Tel: 06121/2051 TWX: 041-86588

Fairchild Halbleiter GmbH
8000 München 2
Bayerstr. 15
Tel: 0811/593632 Telex: 0524831

Fairchild Halbleiter GmbH
3000 Hannover
Königsworther Str. 23
Tel: (0511) 1 78 44 Telex: 092-2922

Fairchild Halbleiter GmbH
7000 Stuttgart-Nord
Parler Strasse 65
Tel: (0711) 22 35 75 Telex: 072-2644

JAPAN
TDK-Fairchild
Sanyo Kokusaku Bldg. 2nd Fl.
7-8 Shiguya 1-Chrome
Shibuya-ku, Tokyo
106, Japan
Tel: (03) 400-8351 TWX: 2424173

MEXICO
Fairchild Mexicana S/A
Blvd. Presidente Adolfo Lopez Mateos 163
Col. Mixcoac, Mexico 19, D.F.
Tel: 563-5411 Telex: 0071-71-038

THE NETHERLANDS
Fairchild Semiconductor
Wal 11
Eindhoven
Tel: 0314067727 Telex: 04451024

SWEDEN
Fairchild Semiconductor AB
Svartensgatan 6,
S-11620 Stockholm
Tel: 08-449255 Telex: 17759

UNITED KINGDOM
Fairchild Semiconductor Ltd.
Kingmaker House, Station Road
New Barnet, Herts EN5 INX
Tel: 0441/4407311 Telex: 051-262 835

FAIRCHILD
SEMICONDUCTOR

©1973 Semiconductor Components Group, Fairchild Camera and Instrument Corporation/464 Ellis Street, Mountain View, California 94042/(415) 962-5011/TWX 910-379-6435
Printed in U.S.A./32-22-0001-072/100M